Definitions of Historic Events for SG6801MSX

Table 1. Event Categories and Codes

Category Name	Category Code
DAT	1
DNP	3
LOG	4
SUM	5
CMD	8
CFM	11
FLT	12
SWX	13
FPX	14
ACW	15
WFC	16
DIM	21
IIM	22
WFM	23
UTL	24
MOR	25
NET	27
EVT	30
RTL	31

Table 2. Types of Additional Data Items for Historic Events (170 items)

<table>
<tr><th>Data Type Name</th><th>Represented As</th><th>Type Information</th></tr>
<tr><td>Action</td><td>UINT16</td><td>Action in action path
<table>
<tr><th colspan="3">Enumeration</th></tr>
<tr><th>Binary</th><th>Value</th><th>Descripion</th></tr>
<tr><td>4</td><td>CloseForXfer</td><td>Close for xfer</td></tr>
<tr><td>20</td><td>ContractRequest</td><td>Contract request</td></tr>
</table></td></tr>
</table>

		21 ContractTerminate Contract Terminate 30 BlockRecloser Block recloser 31 UnblockRecloser Unblock recloser 33 BlockGroundTrip Block ground trip 34 UnblockGroundTrip Unblock ground trip 36 AlternateSettings Alternate settings 37 Normal Normal 200 ActionPathDone Action path done
ActiveURBEClassMask	UINT16	Active URBE Class Mask **Enumeration** **Binary / Value / Descripion** 0 None No URBE 2 Class1 Class 1 active 4 Class2 Class 1 active 6 Class12 Class 1 and 2 active 8 Class3 Class 3 active 10 Class13 Class 1 and 3 active 12 Class23 Class 2 and 3 active 14 Class123 Class 1 and 2 and 3 active
Agent	UINT16	**Numeric Type, Range: 0-65535** **Action / Value** Multiplier 1 Adder 0
Amps	INT16	Show value in Amps **Numeric Type, Range: 0-65535** **Action / Value** Multiplier 1 Adder 0
AnyPhaseLoss	UINT16	Pole X Phase Loss **Numeric Type, Range: 0-15** **Action / Value**

<table>
<tr><td></td><td></td><td><table><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>AverageLoad</td><td>UINT16</td><td>Averaged Load
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>BatteryEvent</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>BatLow</td><td>Battery Low</td></tr><tr><td>2</td><td>BatBad</td><td>Battery Bad</td></tr><tr><td>3</td><td>MonTaskDelayed</td><td>Mon Task Delayed</td></tr><tr><td>4</td><td>StopResched</td><td>Stop and Reschedule</td></tr><tr><td>5</td><td>BadBootTemp</td><td>Bad Boot Temperature</td></tr><tr><td>6</td><td>HeatOn</td><td>Heater On</td></tr><tr><td>7</td><td>HeatOff</td><td>Heater Off</td></tr><tr><td>8</td><td>PowerLoss</td><td>Power Loss</td></tr><tr><td>9</td><td>PowerBack</td><td>Power Back</td></tr><tr><td>10</td><td>StartBatTest</td><td>Start Battery Test</td></tr><tr><td>11</td><td>SwitchTestDelay</td><td>Switch Test Delay</td></tr><tr><td>12</td><td>ChargerSkip</td><td>Charger Skip</td></tr><tr><td>13</td><td>ChargerBad</td><td>Charger Bad</td></tr><tr><td>14</td><td>TestHWError</td><td>Test Hardware Error</td></tr><tr><td>15</td><td>TestDone</td><td>Test Done</td></tr><tr><td>16</td><td>TestInFifteen</td><td>Test In Fifteen</td></tr><tr><td>17</td><td>TestHourly</td><td>Test Hourly</td></tr><tr><td>18</td><td>TestTwoHours</td><td>Test Two Hours</td></tr><tr><td>19</td><td>TestDaily</td><td>Test Daily</td></tr><tr><td>20</td><td>SysError</td><td>System Error</td></tr></table></td></tr>
<tr><td>BMMRESULT</td><td>UINT16</td><td>BMM RESULT
<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr></table></td></tr>
</table>

		0	BMMRESSUCS	BMM Result is Successful
		1	BMMRESBADREQ	BMM Result is bad
		2	BMMRESBUSY	BMM Result is busy
		3	BMMRESBMMUNAVA	BMM Result is unavailable
		4	BMMRESBMMREADFAIL	BMM Result failed to read
		5	BMMRESBMMWRITEFAIL	BMM Result failed to write
		6	BMMRESRENMATCO	BMM Result Read but not matching copy
		7	BMMRESOVERFLOW	BMM Result has overflowed
		8	BMMRESBADRECORD	BMM Result has a bad record
		9	BMMRESNOFON	BMM Result is not found
BMMSTATE	UINT16	BMM STATE		
		Enumeration		
		Binary	**Value**	**Descripion**
		0	BMM_STATE_READY	BMM is ready to process API calls
		1	BMM_STATE_BUSY	BMM is busy processig an API call or with maintenance
		2	BMM_STATE_INIT	BMM is initializing
		3	BMMSTATNOTDETECTED	BMM hardware is not accessible
		4	BMM_STATE_ERROR	A fatal run-time error encountered

<table>
<tr><td></td><td></td><td><table><tr><td></td><td></td><td>during operation</td></tr></table></td></tr>
<tr><td>BooleanTrueFalse</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Description</th></tr><tr><td>0</td><td>false</td><td>false</td></tr><tr><td>1</td><td>true</td><td>true</td></tr></table></td></tr>
<tr><td>bufsiz</td><td>UINT16</td><td><table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>CallbackStatus</td><td>UINT16</td><td>Callback Status<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Description</th></tr><tr><td>0</td><td>ResponseReceived</td><td>Message response received successfully</td></tr><tr><td>1</td><td>NoResponse</td><td>Message received no response</td></tr><tr><td>2</td><td>ExpiredOnXmitList</td><td>Message expired on the transmit list</td></tr><tr><td>3</td><td>ResponseDataError</td><td>Message response data contained an error</td></tr><tr><td>4</td><td>NoAssocPeerDevice</td><td>Message had no associated peer device</td></tr></table></td></tr>
<tr><td>ClearPRBlocker</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Description</th></tr><tr><td>0</td><td>MasterBlks</td><td>Master Blocks</td></tr><tr><td>1</td><td>SetptBlks</td><td>Setpoint Blocks</td></tr></table></td></tr>
<tr><td>CoachRejectionCode</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Description</th></tr><tr><td>1</td><td>Old</td><td>Old</td></tr><tr><td>2</td><td>Duplicate</td><td>Duplicate</td></tr><tr><td>3</td><td>CRCBad</td><td>CRC bad</td></tr></table></td></tr>
<tr><td>CommandCode</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Description</th></tr></table></td></tr>
</table>

		1	OpenSw1	Open Sw1
		2	OpenSw2	Open Sw2
		3	OpenSw3	Open Sw3
		4	CloseSw1	Close Sw1
		5	CloseSw2	Close Sw2
		6	CloseSw3	Close Sw3
		7	BATTest	Start Bat.Test
		8	STLSw1	Shots to Lockout SW1
		9	STLSw2	Shots to Lockout SW2
		10	ClearOC	Clear Overcurrent
		11	FailOvDis	Failure Override Disable
		12	FailOvEna	Failure Override Enable
		13	AutoDis	Auto Op Disable
		14	AutoEna	Auto Op Enable
		15	Auto1Dis	Auto Op Sw1 Disable
		16	Auto1Ena	Auto Op Sw1 Enable
		17	Auto2Dis	Auto Op Sw2 Disable
		18	Auto2Ena	Auto Op Sw2 Enable
		19	HLTDis	HLT Disable
		20	HLTEna	HLT Enable
		21	HLT1Dis	HLT Sw1 Disable
		22	HLT1Ena	HLT Sw1 Enable
		23	HLT2Dis	HLT Sw2 Disable
		24	HLT2Ena	HLT Sw2 Enable
		25	HLT3Dis	HLT Sw3 Disable
		26	HLT3Ena	HLT Sw3 Enable
		27	USER	User-defined
		28	ClrMan	Clear Manual Op
		29	SRBlock	SCADA+Restoration Block
		30	SREna	SCADA+Restoration Enable
		31	SABlock	SCADA+Auto Op Block
		32	SAEna	SCADA+Auto Op Enable
		33	RegCnt	Set Regular Reclose Count
		34	AltCnt	Set Alternate Reclose Count
		35	BlkRest	Broadcast Prohibit Restore

		36 OvRdAutoRev Override op w auto reverse 37 OvRdNoRev Override op w/o auto reverse 38 LvAlignMode Exit Align Mode 39 ResetOpOC Reset Op OC indication
CommandSource	UINT16	**Enumeration** **Binary / Value / Descripion** 0 Undef Undefined 1 AutoOp Automatic Operations 2 SCADA Remote Operations 3 ILink WiFi or other 4 FP Faceplate
CommErrorCode	UINT16	Communications error code **Numeric Type, Range: 0-65535** **Action / Value** Multiplier 1 Adder 0
connectID	UINT16	Connection ID **Numeric Type, Range: 0-65535** **Action / Value** Multiplier 1 Adder 0
ConnectionID	UINT16	ConnectionID **Enumeration** **Binary / Value / Descripion** 0 CIDUndefined CID Undefined 1 CIDGeneral CID General 2 CIDIInternalDNP CID IInternal DNP 3 CIDCoach CID Coach 4 CIDRunner CID Runner 5 CIDContractAgt CID Contract Agent 6 CIDNetlistXfer CID Netlist Xfer 7 CIDCoraAlleyOop CID CoraAlley Oop 8 CIDIT2Events CID IT2 Events

		9 CIDProtection CID Protection 10 CIDDATLoadMgmt CID DAT Load Mgmt 11 CIDCECSignal CID CEC Signal 12 CIDD1ag CID Diag 13 CIDRemoteXmit CID Remote Xmit 16 CIDNETObjMgmt CID NET Object Mgmt 17 CIDNETRunner0 CID NET Runner 0 18 CIDNETRunner1 CID NET Runner 1 19 CIDNETRunner2 CID NET Runner 2 20 CIDNETRunner3 CID NET Runner 3 21 CIDNETRunner4 CID NET Runner 4 22 CIDNETRunner5 CID NET Runner 5 23 CIDNETRunner6 CID NET Runner 6 24 CIDNETRunner7 CID NET Runner 7 25 CIDNETRunner8 CID NET Runner 8 26 CIDNETRunner9 CID NET Runner 9 27 CIDNETRunner10 CID NET Runner 10 28 CIDNETRunner11 CID NET Runner 11 29 CIDNETRunner12 CID NET Runner 12 30 CIDNETRunner13 CID NET Runner 13 31 CIDNETRunner14 CID NET Runner 14 32 CIDNETRunner15 CID NET Runner 15
ContractState	UINT16	Contract State. **Enumeration** **Binary** **Value** **Descripion** 1 CActive Contract state active 2 CReqPend Contract state request pending 3 CReqUnsent Contract state request unsent 4 CReqTrav Contract state request traveling 5 CReqAccept Contract state request accpected 6 CReqDecline Contract state request declined

		7 CReqDeclContd Contract state request decline continuing 8 CDslveStart Contract state dissolve start 9 CDslveContd Contract state dissolve continue 10 CMaintStart Contract state maintenance start 11 CMaintTickl Contract state maintenance tickle 12 CMaintTrav Contract state maintenance travel 13 CMTravNoFind Contract state maintenance travel not found 14 CMTrRetNoFnd Contract state maint travel return not found 15 CMTicklNoFind Contract state maintenance tickle not found 16 CMTiRetNoFnd Contract state maint tickle return not found 17 CMaintReact Contract state maintenance reactivate contract 18 CMNoReact Contract state maint don't reactivate here - continue 19 CChk2ndFld Contract state request - check 2nd field
ControlCode	UINT16	Control Number **Numeric Type, Range: 0-15** **Action** / **Value** Multiplier / 1 Adder / 0
CycleNumber	UINT16	Cycle number **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
DataLength	UINT16	Length of the data

		Numeric Type, Range: 0-65535 **Action** / **Value** Multiplier / 1 Adder / 0
DATARNR	UINT16	Data runnes working within the time limits **Enumeration** **Binary** / **Value** / **Descripion** 1 / NETXDATRNR / Runners are working within timeouts when set
dctsiz	UINT16	Data Length **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
DestinationIndex	UINT16	Destination Index **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
DestNodeID	UINT16	NET Destination Node ID **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
dev	UINT16	Device Type, either a substation or not a substation **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
DeviceIndex	UINT16	local device number of the FN Runner Source Device **Numeric Type, Range: 0-65535**

		Action / **Value** Multiplier / 1 Adder / 0
DeviceNumber	UINT16	Working runner source list in the Runner Source Control **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
DeviceWirepairs	UINT16	Number of Device-Wire pairs from prep function **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
DiagTestType	UINT16	Type of diagnostic test **Enumeration** **Binary** / **Value** / **Descripion** 0 / Scheduled / Scheduled 1 / Quick / Quick
Direction	UINT16	Direction **Enumeration** **Binary** / **Value** / **Descripion** 1 / Forward / Forward 2 / Reverse / Reverse
DivBy10	UINT16	Divide by 10 **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 0.1 Adder / 0
DNPAddress	UINT16	Values represented in hexadecimal code **Numeric Type, Range: 1-65519** **Action** / **Value**

		Multiplier: 1 Adder: 0 **Presentation** **Binary** {0:X2}
DNPAppControl	UINT16	DNP application control byte **Numeric Type, Range: 0-255** **Action** / **Value** Multiplier: 1 Adder: 0
DNPAppFunctionCode	UINT16	DNP Application Function Code **Numeric Type, Range: 0-255** **Action** / **Value** Multiplier: 1 Adder: 0
DNPDLAction	UINT16	DNP DL Action **Enumeration** **Binary** / **Value** / **Descripion** 1 / RecordReinit / record reinit 2 / ResetSeqNumOnly / reset seq num only 3 / NewPeerAdded / new peer added
DNPObjectType	UINT16	DNP Object Type **Numeric Type, Range: 0-255** **Action** / **Value** Multiplier: 1 Adder: 0
DNPObjectVariation	UINT16	DNP Object Variation **Numeric Type, Range: 0-255** **Action** / **Value** Multiplier: 1 Adder: 0
DNPOutputBlockStatusCode	UINT16	DNP Output Block Status Code

		Numeric Type, Range: 0-255 **Action** / **Value** Multiplier / 1 Adder / 0
DNPPointCode	UINT16	Internal DNP point code **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
DNPPointType	UINT16	DNP Point Type **Enumeration** **Binary** / **Value** / **Descripion** 1 / Binary / binary 2 / Analog / analog 3 / Counter / counter
DNPTranspFunctionCode	UINT16	DNP transport function code **Numeric Type, Range: 0-255** **Action** / **Value** Multiplier / 1 Adder / 0
DNPTransportHeader	UINT16	DNP transport header byte **Numeric Type, Range: 0-255** **Action** / **Value** Multiplier / 1 Adder / 0
EnabDisab	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 1 / Enabled / Enabled 0 / Disabled / Disabled
EntryPointDevice	UINT16	Indication that this control is the Entry Point Device **Numeric Type, Range: 0-65535**

		Action / **Value** Multiplier: 1 Adder: 0
ErrorCodeNETX	UINT16	Error code for the Netx system **Enumeration** **Binary** / **Value** / **Descripion** 0 / NETX_SUCCESS / Netx is successful 32 / NETX_BAD_VER / Netx has a bad version 64 / NETX_NO_NETVW / Netx has no Net view 128 / NETX_NO_RTU / No RTU was found for NETX 256 / NETX_BAD_NETVW / Netx has a bad Net view 512 / NETX_OBJ_FAIL / Netx object has failed 1024 / NETX_FAIL / Netx has failed 2048 / NETX_OBJ_QUED / Netx object has been qued 4096 / NETX_NOT_RSD / Netx has not RSD 8192 / NETX_BAD_CRC / Netx has a bad CRC 16384 / NETX_NOTACTV / Netx is not active
ExcessLoad	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 0 / ExcessLoad / Excess Load
ExpressNetViewRow	UINT16	Number of entries in row, one for device,one for wire **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier: 1 Adder: 0
FaultBits	UINT16	Fault Bits **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier: 1 Adder: 0

<table>
<tr><td>FeederNetCRC</td><td>UINT16</td><td>CRC of this Feeder Net
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>FeederNetID</td><td>UINT16</td><td>Feeder Net ID
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>FeederNetIndex</td><td>UINT16</td><td>Feeder Net Index
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>FeederNetObjectFlags</td><td>UINT16</td><td>Incoming Feeder Net Flages from ITD
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>FeederNetObjType</td><td>UINT16</td><td>Type of netlist distribution
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>FixQuality</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>NoSignal</td><td>No Signal</td></tr><tr><td>1</td><td>GPS</td><td>GPS</td></tr><tr><td>2</td><td>DGPS</td><td>DGPS</td></tr><tr><td>3</td><td>PPS</td><td>PPS</td></tr><tr><td>4</td><td>RealTmKinematic</td><td>Real Time Kinematic</td></tr></table></td></tr>
</table>

<table>
<tr><td></td><td></td><td>
<table>
<tr><td>5</td><td>FloatRTK</td><td>Float RTK</td></tr>
<tr><td>6</td><td>Estimated</td><td>Estimated</td></tr>
<tr><td>7</td><td>MnlInputMode</td><td>Manual Input Mode</td></tr>
<tr><td>8</td><td>SimulatMode</td><td>Simulation Mode</td></tr>
</table>
</td></tr>
<tr><td>FreqTripType</td><td>UINT16</td><td>Over Under Freq. Trip
<table>
<tr><th colspan="2">Numeric Type, Range: 1-2</th></tr>
<tr><th>Action</th><th>Value</th></tr>
<tr><td>Multiplier</td><td>1</td></tr>
<tr><td>Adder</td><td>0</td></tr>
</table>
</td></tr>
<tr><td>GPSStatus</td><td>UINT16</td><td>
<table>
<tr><th colspan="3">Enumeration</th></tr>
<tr><th>Binary</th><th>Value</th><th>Descripion</th></tr>
<tr><td>0</td><td>TimePosOK</td><td>Time Position OK</td></tr>
<tr><td>1</td><td>TPosOKPendStab</td><td>Time Position OK Pending Stability</td></tr>
<tr><td>2</td><td>TPosOKNoPulse</td><td>Time Position OK No Pulse Signal</td></tr>
<tr><td>3</td><td>TimePosInvalid</td><td>Time Position Invalid</td></tr>
<tr><td>4</td><td>DInvalidSLineOdd</td><td>Data Invalid Serial Line Anomaly</td></tr>
<tr><td>5</td><td>DInvalidSLineErr</td><td>Data Invalid Serial Line Error</td></tr>
<tr><td>6</td><td>DInvalidSLineSil</td><td>Data Invalid Serial Line Silence</td></tr>
<tr><td>7</td><td>DisabledByUser</td><td>Disabled By User</td></tr>
<tr><td>255</td><td>NotInitialized</td><td>Not Initialized</td></tr>
</table>
</td></tr>
<tr><td>HealingType</td><td>UINT16</td><td>
<table>
<tr><th colspan="3">Enumeration</th></tr>
<tr><th>Binary</th><th>Value</th><th>Descripion</th></tr>
<tr><td>1</td><td>ITII</td><td>IT-II</td></tr>
<tr><td>2</td><td>RSH</td><td>RSH</td></tr>
</table>
</td></tr>
<tr><td>Hexushort</td><td>UINT16</td><td>Values represented in hexadecimal code
<table>
<tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr>
<tr><th>Action</th><th>Value</th></tr>
<tr><td>Multiplier</td><td>1</td></tr>
<tr><td>Adder</td><td>0</td></tr>
</table>
</td></tr>
</table>

		Presentation **Binary** {0:X2}
HLT	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 0 / HLTInactive / HLT Inactive 1 / HLTActive / HLT Active
INCNUMTEAM	UINT16	FEEDERNET incoming team array number of teams **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
IndexMngTable	UINT16	Find this node in the NLTA arrays **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
Instance	UINT16	Instance: The value is used for debugging purposes and does not yield any benefit **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
InternalCode	UINT16	Internal event-specific code **Enumeration** **Binary** / **Value** / **Descripion** 1 / CCircRsrcOK / Contract circuit resources are adequate 2 / CLineSegReqHi / Contract line segment request above limit 3 / CLoadCapReqHi / Contract load capacity request above limit 4 / SrcSubSwVloss / SourceSub Switch has detected voltage loss

<table>
<tr><td>LineSegLimit</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>LineSegLimit</td><td>Line Seg Limit</td></tr></table></td></tr>
<tr><td>LoadingCurrent</td><td>UINT16</td><td>1 amperes per count
<table><tr><th colspan="2">Numeric Type, Range: 0-655350</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>LOGDiagType</td><td>UINT16</td><td>LOG Diagnosos Type
<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>Alarm</td><td>Alarm</td></tr><tr><td>2</td><td>Warning</td><td>Warning</td></tr><tr><td>4</td><td>Error</td><td>Error</td></tr></table></td></tr>
<tr><td>ManualOp</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>ManOpInactive</td><td>Man Op Inactive</td></tr><tr><td>1</td><td>ManOpActive</td><td>Man Op Active</td></tr></table></td></tr>
<tr><td>MapNumber</td><td>UINT16</td><td>MapNumber
<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>BinaryInput</td><td>binary input</td></tr><tr><td>1</td><td>AnalogInput</td><td>analog input</td></tr><tr><td>2</td><td>Counter</td><td>counter</td></tr><tr><td>3</td><td>Control</td><td>control</td></tr><tr><td>4</td><td>AnalogOutput</td><td>analog output</td></tr><tr><td>5</td><td>DblBinaryInput</td><td>double binary output</td></tr></table></td></tr>
<tr><td>Master</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>Master1</td><td>Master 1</td></tr><tr><td>2</td><td>Master2</td><td>Master 2</td></tr><tr><td>3</td><td>Master3</td><td>Master 3</td></tr></table></td></tr>
</table>

		4 Master4 Master 4 5 Master5 Master 5 6 Master6 Master 6 255 NoMaster No Master
MasterRecAddError	UINT16	**Enumeration** **Binary** **Value** **Descripion** 0 Good good 1 Error error 2 AlrdyInList already in list 3 ListFull list full 4 NotOnList not on list
MaxPeersWhen	UINT16	Max Peers Occur During This **Enumeration** **Binary** **Value** **Descripion** 0 WhenTestInit WhenTestInit 1 WhenTestStats WhenTestStats
MemberWithDoneAndEOL	UINT16	**Enumeration** **Binary** **Value** **Descripion** 1 Member1 Member 1 2 Member2 Member 2 3 Member3 Member 3 4 Member4 Member 4 5 Member5 Member 5 6 Member6 Member 6 7 Member7 Member 7 8 Member8 Member 8 254 EndofList EndofList 256 Done Done
MoreThanOneDiffFound	UINT16	**Enumeration** **Binary** **Value** **Descripion** 0 OneDiffFound Only OneDiff Found 1 TwoOrMoreDiffs Two Or More Diffs
MotorOperation	UINT16	**Enumeration**

		Binary / **Value** / **Descripion** 0 / NoChange / No Change 1 / Closed / Switch Closed 2 / Open / Switch Open 3 / FindClose / Torque Close and Set Close Limit 4 / ClswTorque / Torque Open then Close 5 / JogClose / Jog Switch Close 6 / JogOpen / Jog Switch Open 7 / OpenwTorque / Torque Close then Open 8 / OverrideOpen / Override Open 9 / OverrideClose / Override Close 10 / SetClose / Set Close Limit via PC 11 / SetOpen / Set Open Limit via PC 12 / OvrdOpenNoRev / Override Open with No Reverse 13 / OvrdClsNoRev / Override Close with No Reverse 14 / FindOpen / Torque Open and Set Open Limit 16 / EncodDriftLog / Encoder Drift Logging
MotorOpReason	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 0 / NoOp / No Operation 16 / FrontPanelOp / Front Panel Operation 32 / SCADAOp / SCADA Operation 48 / TestOp / Test Operation 64 / AutoOp / Automatic Operation 80 / EmergencyRevOp / Emergency Reverse Operation 96 / RemoteCCOp / Remote Contact Close Operation 128 / PCOp / PC Operation 160 / ErrorOp / Error Operation 176 / UnknownedReq / Unknown Request

<table>
<tr><td>netfragID</td><td>UINT16</td><td>Net Fragment ID
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>NETLISTRNR</td><td>UINT16</td><td>Number of NETLIST runners in RunnerSource
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>netobuffFree</td><td>UINT16</td><td>Cumulative free buffer space
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>NETSTE</td><td>UINT16</td><td>Net state E
<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>BACKIDLE</td><td>Go back to idle</td></tr><tr><td>8</td><td>RESENDRNR</td><td>Go re-send the runner objects since some failed</td></tr><tr><td>4</td><td>WAITRT</td><td>Go wait for returns</td></tr></table></td></tr>
<tr><td>NetViewAnalyzer</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>NETVIEWNOTVALID</td><td>Net view is not valid</td></tr><tr><td>2</td><td>NETVIEWVALID</td><td>Net view is valid</td></tr></table></td></tr>
<tr><td>NETViewLength</td><td>UINT16</td><td>The Normal State NetView Length of each row table
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
</table>

<table>
<tr><td>NetViewRowNum</td><td>UINT16</td><td>Runner source row number in nnet
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>NETVSRC</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>NETVSRC_NORMAL</td><td>NETV source is normal</td></tr><tr><td>2</td><td>NETVSRC_ALTERNATE</td><td>NETV source is alternate</td></tr><tr><td>3</td><td>NETVFDR_NOTNORMAL</td><td>NETV source is not normal</td></tr><tr><td>255</td><td>NETVSRC_UNDEFINED</td><td>NETV source is undefined</td></tr></table></td></tr>
<tr><td>NodeCount</td><td>UINT16</td><td>Number of appearnaces in this runner
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>NodeIndex</td><td>UINT16</td><td>Our position in the Node Index table
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>Nodelists</td><td>UINT16</td><td>Runner node lists
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>NoMaster</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>MasterExists</td><td>MasterExists</td></tr></table></td></tr>
</table>

		1 NoMaster No Master
NoPeersWhen	UINT16	Max Peers Occur During This **Enumeration** **Binary / Value / Descripion** 0 SchedTest SchedTest 1 QuickTest QuickTest 2 KeepAlive KeepAlive 3 KwikKeepAlive KwikKeepAlive
nsngl	UINT16	number of device connections present, counter, list len **Numeric Type, Range: 0-65535** **Action / Value** Multiplier 1 Adder 0
nt	UINT16	Number of team records **Numeric Type, Range: 0-65535** **Action / Value** Multiplier 1 Adder 0
NumberFrames	UINT16	Number frames have been currently written on CF **Numeric Type, Range: 0-3600** **Action / Value** Multiplier 1 Adder 0
NumDataElements	UINT16	Actual number of data elements in DLV **Numeric Type, Range: 0-65535** **Action / Value** Multiplier 1 Adder 0
NumDevicesinFN	UINT16	Number of devices in the Feeder Net **Numeric Type, Range: 0-65535** **Action / Value**

		Multiplier: 1 Adder: 0
NumNodeID	UINT16	Number of Node IDs in the list **Numeric Type, Range: 0-65535** Multiplier: 1 Adder: 0
NumNodeIndex	UINT16	Number of indexes in the Node table **Numeric Type, Range: 0-65535** Multiplier: 1 Adder: 0
NumOfDevices	UINT16	Netview number of devices/switches in DivisionNet **Numeric Type, Range: 0-65535** Multiplier: 1 Adder: 0
NumOfPowSrc	UINT16	Number of power sources in DivisionNet **Numeric Type, Range: 0-65535** Multiplier: 1 Adder: 0
NumOfRows	UINT16	Net Present NetView number of rows in table for DivisionNet **Numeric Type, Range: 0-65535** Multiplier: 1 Adder: 0
NumOfTeams	UINT16	Number of teams **Numeric Type, Range: 0-65535** Multiplier: 1

		Adder: 0
NumOfWirePairs	UINT16	Number of entries in the D-W (Device-Wire) tables **Numeric Type, Range: 0-65535** Action: Multiplier, Value: 1 Adder: 0
NumPaths	UINT16	Number of paths in this particular Feeder Net **Numeric Type, Range: 0-65535** Action: Multiplier, Value: 1 Adder: 0
NumPayLoad	UINT16	Quantity of associated payload elements **Numeric Type, Range: 0-65535** Action: Multiplier, Value: 1 Adder: 0
NumRunners	UINT16	Number of runners **Numeric Type, Range: 0-65535** Action: Multiplier, Value: 1 Adder: 0
ObjectLength	UINT16	**Numeric Type, Range: 0-65535** Action: Multiplier, Value: 1 Adder: 0
ObjID	UINT16	Object ID **Numeric Type, Range: 0-65535** Action: Multiplier, Value: 1 Adder: 0
ObjState	UINT16	**Enumeration**

		<table><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>NETOBJSTATENONE</td><td>No net object state</td></tr><tr><td>1</td><td>NETOBJSTATEREADY</td><td>Net object state is ready</td></tr><tr><td>2</td><td>NETOBJSEBUFFERED</td><td>Net object state is buffered</td></tr><tr><td>3</td><td>NETOBJSTATEACKEXEC</td><td>Net object state positive ack</td></tr><tr><td>4</td><td>NETOBJNACKEXEC</td><td>Net object state negative ack</td></tr><tr><td>5</td><td>NETOBJSTATEHOLD</td><td>Net object state is on hold</td></tr><tr><td>6</td><td>NETOBJSTATESEND</td><td>Net object state sent</td></tr><tr><td>7</td><td>NETOBJSENDWAIT</td><td>Net object state is waiting to be sent</td></tr><tr><td>8</td><td>NETOBJSENDCONTINUE</td><td>Net object state is contiued to be sent</td></tr><tr><td>9</td><td>NETOBJSTATEREMOVE</td><td>Net object state is removed</td></tr><tr><td>10</td><td>NETOBJRECEIVED</td><td>Net object state is recieved</td></tr><tr><td>11</td><td>NETOBJSENDDELAY</td><td>Net object state is delayed</td></tr></table>
OCEndSeconds	INT16	Show value in Seconds <table><tr><th colspan="2">Numeric Type, Range: 0.001-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>0.00625</td></tr><tr><td>Adder</td><td>0</td></tr></table>
OneHexByte	UINT16	Values represented in hexadecimal code <table><tr><th colspan="2">Numeric Type, Range: 0-255</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table>

<table>
<tr><td></td><td></td><td><table><tr><th colspan="2">Presentation</th></tr><tr><td>Binary</td><td>{0:X2}</td></tr></table></td></tr>
<tr><td>Point</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>OperateIR</td><td>Point to operate IR</td></tr><tr><td>2</td><td>BlockReclosing</td><td>Point to block reclosing</td></tr><tr><td>3</td><td>BlockGroundTrip</td><td>Point to block ground trip</td></tr><tr><td>4</td><td>ChangeProfile</td><td>Point to change profile.</td></tr></table></td></tr>
<tr><td>PortCode</td><td>UINT16</td><td>Communication port code.<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>UDP</td><td>UDP</td></tr><tr><td>1</td><td>PortA</td><td>portA</td></tr><tr><td>2</td><td>PortB</td><td>portB</td></tr><tr><td>3</td><td>PortC</td><td>portC</td></tr><tr><td>4</td><td>PortD</td><td>portD</td></tr><tr><td>6</td><td>TCPV6</td><td>TCP Virtual Port 6</td></tr><tr><td>7</td><td>TCPV7</td><td>TCP Virtual Port 7</td></tr><tr><td>8</td><td>TCPV8</td><td>TCP Virtual Port 8</td></tr><tr><td>9</td><td>TCPV9</td><td>TCP Virtual Port 9</td></tr><tr><td>10</td><td>TCPV10</td><td>TCP Virtual Port 10</td></tr><tr><td>11</td><td>TCPV11</td><td>TCP Virtual Port 11</td></tr><tr><td>12</td><td>TCPV12</td><td>TCP Virtual Port 12</td></tr><tr><td>13</td><td>TCPV13</td><td>TCP Virtual Port 13</td></tr><tr><td>14</td><td>TCPV14</td><td>TCP Virtual Port 14</td></tr><tr><td>15</td><td>TCPV15</td><td>TCP Virtual Port 15</td></tr><tr><td>16</td><td>TCPV16</td><td>TCP Virtual Port 16</td></tr><tr><td>17</td><td>TCPV17</td><td>TCP Virtual Port 17</td></tr><tr><td>18</td><td>TCPV18</td><td>TCP Virtual Port 18</td></tr><tr><td>19</td><td>TCPV19</td><td>TCP Virtual Port 19</td></tr><tr><td>20</td><td>TCPV20</td><td>TCP Virtual Port 20</td></tr><tr><td>21</td><td>TCPV21</td><td>TCP Virtual Port 21</td></tr></table></td></tr>
</table>

		22 TCPV22 TCP Virtual Port 22 23 TCPV23 TCP Virtual Port 23 24 TCPV24 TCP Virtual Port 24 25 TCPV25 TCP Virtual Port 25 26 TCPV26 TCP Virtual Port 26 27 TCPV27 TCP Virtual Port 27 28 TCPV28 TCP Virtual Port 28 29 TCPV29 TCP Virtual Port 29 30 TCPV30 TCP Virtual Port 30 31 TCPV31 TCP Virtual Port 31 32 TCPV32 TCP Virtual Port 32 33 TCPV33 TCP Virtual Port 33 34 TCPV34 TCP Virtual Port 34 35 TCPV35 TCP Virtual Port 35 36 TCPV36 TCP Virtual Port 36 37 TCPV37 TCP Virtual Port 37 38 TCPV38 TCP Virtual Port 38 255 Undefined Undefined 20000 P20000 20000 20001 P20001 20001
PRReason	UINT16	**Enumeration** **Binary** **Value** **Descripion** 1 UnderFrequency Under Frequency 2 HotLineTag Hot Line Tag 3 ManualOperation Manual Operation 4 LocalCommand Local Command 5 AuxInput Aux Input 6 StateMismatch State Mismatch 7 SCADAMaster SCADA Master
RegistrationStatus	UINT16	**Enumeration** **Binary** **Value** **Descripion** 0 DNPRETURNGOOD DNP return is good 1 DNPRETURNERROR DNP return error

		2	DNPRTNERRONLIST	DNP return error already on the list
		3	DNPRTNERRLISTFULL	DNP return error is full
		4	DNPRTNOTONLIST	DNP return error not on list
		5	DNPRTNERRNOTP2P	DNP return error not P2P
		6	DNPRTNDELAY	DNP return delayed before retry
		7	DNPRTNACTIVEPOINT	DNP return active on point
		8	DNPRETURNCOMPLETE	DNP return is complete
		9	DNPRTNINPROGRESS	DNP return in progress
		10	DNPRRNNOTAMASTER	DNP return not a master
RestoreConditionTestResult	UINT16	Restore Condition Test Result		

Enumeration		
Binary	**Value**	**Descripion**
0	NoResult	NO RESULT
1	CloseSwOK	CLOSE SW OK
2	Isolated	ISOLATED
3	ExcessLoad	EXCESSIVE LOAD
4	CloseOnly	CLOSE ONLY
5	OpenOnly	OPEN ONLY
6	LineSegLimit	LINE SEGMENT LIMIT
7	NbrProblem	NEIGHBOR ERROR
8	PhaseLoss	PHASE LOSS
9	SrcBrkIsol	SRC BREAK ISOLATED
10	LoadShed	LOAD SHED
11	Short2Srces	SHORT TWO SOURCES
100	CoordProblem	COORD ERROR
101	PrevBadV	PREV BAD VOLT

<table>
<tr><td>RouteEntryError</td><td>UINT16</td><td>Route Entry Error
<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>BrdcastAddrInvalid</td><td>broadcast addresses invalid in routing tables</td></tr><tr><td>2</td><td>IPAddrPortInvalid</td><td>IP address and port indication must be mutually exclusive</td></tr></table></td></tr>
<tr><td>RSD</td><td>UINT16</td><td>Runner Source Device
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>RTUAddress</td><td>UINT16</td><td>DNP address
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>RTUlists</td><td>UINT16</td><td>Runner RTU lists
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>RunnerIndex</td><td>UINT16</td><td>Local runner index number
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>RunnerNumber</td><td>UINT16</td><td>Runner Number
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr></table></td></tr>
</table>

<table>
<tr><td></td><td></td><td><table><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>RunnerQuantity</td><td>UINT16</td><td>Next data collection runner's quantity
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>RunnerSize</td><td>UINT16</td><td>Size of recieved runner in BYTES
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>RunnerType</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>NETOBJTYPEUNDEF</td><td>NetObject type is undefined</td></tr><tr><td>1</td><td>NETOBJTYPEACKMESG</td><td>NetObject type acknowldege message</td></tr><tr><td>2</td><td>NETOBJTYPENACKMESG</td><td>NetObject type not acknowledging message</td></tr><tr><td>3</td><td>NETOBJTYPENETLIST</td><td>NetObject type net list</td></tr><tr><td>4</td><td>NETOBJTYPERAWPUSH</td><td>NetObject type raw push</td></tr><tr><td>16</td><td>NETxCMMCKRSP</td><td>ITD Communications check runner with routing</td></tr><tr><td>19</td><td>NETxNLRQST</td><td>Request a copy of the present Primary Netlist</td></tr><tr><td>21</td><td>NETxNPRIRSP</td><td>Existing Primary Netlist Response to rqst</td></tr></table></td></tr>
</table>

<table>
<tr><td></td><td></td><td>
<table>
<tr><td>22</td><td>NETxACTPNL</td><td>Activate Primary Netlist After Delay</td></tr>
<tr><td>23</td><td>NETxNADJNLD</td><td>New Adjacent Netlist delivery</td></tr>
<tr><td>24</td><td>NETxDATCLCT</td><td>Data Collect Runner</td></tr>
<tr><td>25</td><td>NETxDATDLV</td><td>Data Block delivery runner</td></tr>
<tr><td>27</td><td>NETxADJNLRPT</td><td>Tie delivers an adjacent NL to a Runner Source for delivery</td></tr>
<tr><td>28</td><td>NETxADJDBRPT</td><td>Tie delivers an adjacent DB to a Runner Source for delivery</td></tr>
<tr><td>29</td><td>NETxADJDBDLV</td><td>Runner Source delivers an adjacent Data Block to FN</td></tr>
<tr><td>30</td><td>NETxPEERCHK</td><td>ITD Peer check runner with routing</td></tr>
<tr><td>160</td><td>NETxNETLRPT</td><td>Active Netlist reporter</td></tr>
</table>
</td></tr>
<tr><td>SBOTimerValue</td><td>UINT16</td><td>Timer value for Select-Before-Operate. Unit = 0.1 sec.
<table>
<tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr>
<tr><th>Action</th><th>Value</th></tr>
<tr><td>Multiplier</td><td>1</td></tr>
<tr><td>Adder</td><td>0</td></tr>
</table>
</td></tr>
<tr><td>SCADACommand</td><td>UINT16</td><td>
<table>
<tr><th colspan="3">Enumeration</th></tr>
<tr><th>Binary</th><th>Value</th><th>Descripion</th></tr>
<tr><td>0</td><td>None</td><td>None</td></tr>
<tr><td>1</td><td>PulseOn</td><td>PULSE ON</td></tr>
<tr><td>2</td><td>PulseOff</td><td>PULSE OFF</td></tr>
<tr><td>3</td><td>LatchOn</td><td>LATCH ON</td></tr>
<tr><td>4</td><td>LatchOff</td><td>LATCH OFF</td></tr>
</table>
</td></tr>
</table>

		65 Close CLOSE 129 Trip TRIP
SCADACommandCode	UINT16	**Enumeration** **Binary / Value / Descripion** 8192 BATTest Start Battery Test 8464 OperateSw1 Operate Sw1 8465 OperateSw2 Operate Sw2 8466 OperateSw3 Operate Sw3 8704 STLSw1 Shots to Lockout SW1 8705 STLSw2 Shots to Lockout SW2 8706 ClearOC Clear Overcurrent 8960 FailOvEnaorDis Failure Override Enable/Disable 8962 AutoEnaorDis Auto Op Enable/Disable 8964 Auto1EnaorDis Auto Op Sw1 Enable/Disable 8966 Auto2EnaorDis Auto Op Sw2 Enable/Disable 8968 HLTEnaorDis HLT Disable 8970 HLT1EnaorDis HLT Sw1 Enable/Disable 8972 HLT2EnaorDis HLT Sw2 Enable/Disable 8974 HLT3EnaorDis HLT Sw3 Enable/Disable 8976 AckADI1 Ack ADI 1 8978 AckADI2 Ack ADI 2 8980 AckADI3 Ack ADI 3 9216 OvRdAutoRev Override op w auto reverse 9217 OvRdNoRev Override op w/o auto reverse 9218 LvAlignMode Exit Align Mode 9219 ResetOpOC Reset Op OC indication 9472 ClrErrors Clear Errors 9474 ClrWarnings Clear Warnings 9476 ClrAlarms Clear Alarms
SecondaryEvent	UINT16	Secondary Event Flag for WFC capture **Numeric Type, Range: 0-1** **Action / Value** Multiplier 1

<table>
<tr><td></td><td></td><td><table><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>short</td><td>INT16</td><td>INT16
<table><tr><th colspan="2">Numeric Type, Range: -32768-32767</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>SideInfo</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>SIDEUNKOWN</td><td>Side information is unkown</td></tr><tr><td>1</td><td>SIDE1</td><td>This team member is on side 1 of the switch</td></tr><tr><td>2</td><td>SIDE2</td><td>This team member is on side of the switch</td></tr></table></td></tr>
<tr><td>SizeOfObject</td><td>UINT16</td><td>Size of the arriving object in bytes, also a flag/signal of arrival to State machine A
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>SrcNodeID</td><td>UINT16</td><td>NET Source Node ID
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>State</td><td>UINT16</td><td>State of the transition
<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>TransON</td><td>Transition to On</td></tr><tr><td>2</td><td>TransOFF</td><td>Transition to Off</td></tr></table></td></tr>
<tr><td>StatusFeedbackArea</td><td>UINT16</td><td>Netx status feedback area, Data 1 is current state while Data 2 is the previous state
<table><tr><th colspan="2">Numeric Type, Range: 0-65536</th></tr><tr><th>Action</th><th>Value</th></tr></table></td></tr>
</table>

<table>
<tr><td></td><td></td><td><table><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>Substation</td><td>UINT16</td><td>Present Substation

<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>SUMSessionEndReason</td><td>UINT16</td><td>SUM Session End Reason

<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>DeviceStartup</td><td>Device Startup</td></tr><tr><td>1</td><td>Logoff</td><td>Logoff</td></tr><tr><td>2</td><td>Timeout</td><td>Timeout</td></tr><tr><td>3</td><td>Preempted</td><td>Preempted</td></tr></table></td></tr>
<tr><td>SW1LostPhases</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>None</td><td>None</td></tr><tr><td>2</td><td>A</td><td>A</td></tr><tr><td>4</td><td>B</td><td>B</td></tr><tr><td>6</td><td>AB</td><td>A and B</td></tr><tr><td>8</td><td>C</td><td>C</td></tr><tr><td>10</td><td>AC</td><td>A and C</td></tr><tr><td>12</td><td>BC</td><td>B and C</td></tr><tr><td>14</td><td>ABC</td><td>A and B and C</td></tr></table></td></tr>
<tr><td>SW1Or2</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>SW1</td><td>SW1</td></tr><tr><td>2</td><td>SW2</td><td>SW2</td></tr></table></td></tr>
<tr><td>SW2LostPhases</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>None</td><td>None</td></tr><tr><td>2</td><td>D</td><td>D</td></tr></table></td></tr>
</table>

		4 E E 6 DE D and E 8 F F 10 DF D and F 12 EF E and F 14 DEF D and E and F
SwitchNum	UINT16	**Enumeration** **Binary Value Descripion** 1 Switch1 Switch 1 2 Switch2 Switch 2 3 Switch3 Switch 3
SwitchOperationResult	UINT16	Switch Operation Result **Enumeration** **Binary Value Descripion** 0 SOKNoRes SW OK (no result yet) 1 ClsVerif SW CLOSE VERIFIED 2 ClVerErr SW CLOSE VERIFY FAIL 3 ClVerErrTr SW CLOSE VERIFY FAIL TRIP 4 BadSwNum BAD SW NUM 5 SwNotAuto SW NOT AUTO 6 SBadDir SW BAD DIRECTION 7 SwWasCls SW WAS CLOSED 8 OpnVerif SW OPEN VERIFIED 9 OpnVerErr SW OPEN VERIFY FAIL 10 OpnXcesLd SW OPEN EXCESS LOAD 11 SwWasOpn SW WAS OPEN 12 CmdVerif COMMAND VERIFIED 13 CmdRedun COMMAND REDUNDANT 14 CmdVerErr COMMAND VERIFY FAIL 15 BadCtlPt BAD CTRL PT 16 ClsErrV SW CLOSE FAIL VOLT
SwitchPosition	UINT16	Switch position **Numeric Type, Range: 1-8**

<table>
<tr><td></td><td></td><td>
<table>
<tr><th>Action</th><th>Value</th></tr>
<tr><td>Multiplier</td><td>1</td></tr>
<tr><td>Adder</td><td>0</td></tr>
</table>
</td></tr>
<tr><td>SwitchRecord</td><td>UINT16</td><td>
<table>
<tr><th colspan="2">Numeric Type, Range: 0-8</th></tr>
<tr><th>Action</th><th>Value</th></tr>
<tr><td>Multiplier</td><td>1</td></tr>
<tr><td>Adder</td><td>0</td></tr>
</table>
</td></tr>
<tr><td>SwitchState</td><td>UINT16</td><td>Switch State
<table>
<tr><th colspan="3">Enumeration</th></tr>
<tr><th>Binary</th><th>Value</th><th>Descripion</th></tr>
<tr><td>0</td><td>DATUSWSTAT_UNKOWN</td><td>Switch State is unknown</td></tr>
<tr><td>1</td><td>DATUSWSTAT_CLOSED</td><td>Switch State is closed</td></tr>
<tr><td>2</td><td>DATUSWSTAT_OPEN</td><td>Switch State is open</td></tr>
<tr><td>3</td><td>DATUSWSTAT_TRANS</td><td>Switch State is in transition</td></tr>
<tr><td>4</td><td>DATUSWSTAT_UNDEF</td><td>Switch State is undefined</td></tr>
</table>
</td></tr>
<tr><td>SWnum</td><td>UINT16</td><td>
<table>
<tr><th colspan="3">Enumeration</th></tr>
<tr><th>Binary</th><th>Value</th><th>Descripion</th></tr>
<tr><td>0</td><td>SW1</td><td>SW1</td></tr>
<tr><td>1</td><td>SW2</td><td>SW2</td></tr>
<tr><td>2</td><td>SW3</td><td>SW3</td></tr>
</table>
</td></tr>
<tr><td>SWOp</td><td>UINT16</td><td>
<table>
<tr><th colspan="3">Enumeration</th></tr>
<tr><th>Binary</th><th>Value</th><th>Descripion</th></tr>
<tr><td>1</td><td>Close</td><td>Close</td></tr>
<tr><td>2</td><td>Open</td><td>Open</td></tr>
</table>
</td></tr>
<tr><td>SWPos</td><td>UINT16</td><td>
<table>
<tr><th colspan="3">Enumeration</th></tr>
<tr><th>Binary</th><th>Value</th><th>Descripion</th></tr>
<tr><td>1</td><td>SW1</td><td>SW1</td></tr>
<tr><td>2</td><td>SW2</td><td>SW2</td></tr>
</table>
</td></tr>
</table>

		3 SW3 SW3 4 SW4 SW4 5 SW5 SW5 6 SW6 SW6 7 SW7 SW7 8 SW8 SW8
SWSWOp	UINT16	**Enumeration** **Binary** **Value** **Descripion** 1 SwClose SwClose 2 SwOpen SwOpen
syncstat	UINT16	**Enumeration** **Binary** **Value** **Descripion** 0 DATASYNCHSUCCESS Operation was successful 1 DATSYNCHBADSTATE Present States of Local Switches do not match Normal states 2 DATSYNCHNOTQUIET IntelliTEAM is not 'quiet' (in the process of transferring) 3 DATSYNCHNETBADDATA Netlist data is somehow invalid (e.g. no local teams) 4 DATSYNCHNINPROCESS Presently processing a new netlist 5 DATSYNCHNORTUADDR Device is not configured with an RTU address
SystemVoltage	UINT16	**Enumeration** **Binary** **Value** **Descripion** 1 V69000 69000 2 V46000 46000

		3 V44000 44000 4 V40600 40600 5 V36000 36000 6 V34500 34500 7 V27600 27600 8 V26400 26400 9 V24900 24900 10 V23900 23900 11 V22900 22900 12 V20800 20800 13 V14400 14400 14 V13800 13800 15 V13200 13200 16 V12470 12470 17 V12000 12000 18 V8320 8320 19 V7200 7200 20 V4800 4800 21 V4160 4160 22 V33000 33000 23 V22000 22000 24 V20000 20000 25 V11000 11000 26 V10000 10000
Task	UINT16	**Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
TCPPortActiveState	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 0 / Undefined / Undefined 1 / Unused / Unused 2 / Active / Active

		3 Connect Connect 4 Disconnect Disconnect 5 Reconnect Reconnect 6 Error Error 7 Blocking Blocking 8 KpAliveSend Send Keep Alive
TCPPortConnectState	UINT16	**Enumeration** **Binary / Value / Descripion** 1 Removed Removed 2 Disconnecting Disconnecting 3 Connecting Connecting 4 Changed Changed 5 Badstate Bad state 6 KpAliveSent Keep Alive Sent 7 KpAliveBad Keep Alive Bad
Team	UINT16	Team number **Numeric Type, Range: 1-8** **Action / Value** Multiplier 1 Adder 0
TeamNumber	UINT16	**Enumeration** **Binary / Value / Descripion** 1 Team1 Team 1 2 Team2 Team 2 3 Team3 Team 3 4 Team4 Team 4 5 Team5 Team 5 6 Team6 Team 6 7 Team7 Team 7 8 Team8 Team 8
TeamNumberWithUnknown	UINT16	**Enumeration** **Binary / Value / Descripion** 0 DirUnknown Dir Unknown

		1 Team1 Team 1 2 Team2 Team 2 3 Team3 Team 3 4 Team4 Team 4 5 Team5 Team 5 6 Team6 Team 6 7 Team7 Team 7 8 Team8 Team 8 9 TeamIsVirtual Team is virtual
TeamRecord	UINT16	Team record **Numeric Type, Range: 0-8** **Action** / **Value** Multiplier / 1 Adder / 0
TeamWithDone	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 1 Member1 Member 1 2 Member2 Member 2 3 Member3 Member 3 4 Member4 Member 4 5 Member5 Member 5 6 Member6 Member 6 7 Member7 Member 7 8 Member8 Member 8 256 Done Done
TimeSource	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 1 MCUInternal MCU internal oscillator 2 RTC MCU RTC 3 GPS MCU GPS
totallen	UINT16	Total Length **Numeric Type, Range: 0-65535** **Action** / **Value**

<table>
<tr><td></td><td></td><td><table><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>TransferState</td><td>UINT16</td><td><table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>0</td><td>IDLE</td><td>IDLE</td></tr><tr><td>1</td><td>INIT</td><td>INIT</td></tr><tr><td>2</td><td>RQST</td><td>RQST</td></tr><tr><td>3</td><td>GRNT</td><td>GRNT</td></tr><tr><td>4</td><td>RTN0</td><td>RTN0</td></tr><tr><td>5</td><td>RTN1</td><td>RTN1</td></tr><tr><td>6</td><td>RTN2</td><td>RTN2</td></tr><tr><td>7</td><td>RTN3</td><td>RTN3</td></tr><tr><td>8</td><td>WAIT</td><td>WAIT</td></tr><tr><td>9</td><td>STOP</td><td>STOP</td></tr><tr><td>10</td><td>FAULT</td><td>FAULT</td></tr><tr><td>11</td><td>HOLD</td><td>HOLD</td></tr><tr><td>12</td><td>CORA</td><td>CORA</td></tr><tr><td>13</td><td>PLI</td><td>PLI</td></tr><tr><td>14</td><td>PLIDONE</td><td>PLIDONE</td></tr><tr><td>15</td><td>PLIWAIT</td><td>PLIWAIT</td></tr><tr><td>16</td><td>PLIWAIT2</td><td>PLIWAIT2</td></tr><tr><td>255</td><td>UNDEF</td><td>UNDEF</td></tr></table></td></tr>
<tr><td>URBEStatus</td><td>UINT16</td><td>URBE Status
<table><tr><th colspan="3">Enumeration</th></tr><tr><th>Binary</th><th>Value</th><th>Descripion</th></tr><tr><td>1</td><td>Enabled</td><td>enabled</td></tr><tr><td>0</td><td>Disabled</td><td>disabled</td></tr></table></td></tr>
<tr><td>ushort</td><td>UINT16</td><td>UINT16
<table><tr><th colspan="2">Numeric Type, Range: 0-65535</th></tr><tr><th>Action</th><th>Value</th></tr><tr><td>Multiplier</td><td>1</td></tr><tr><td>Adder</td><td>0</td></tr></table></td></tr>
<tr><td>VAddr16</td><td>UINT16</td><td>16-bit virtual memory address</td></tr>
</table>

		Numeric Type, Range: 0-65535 **Action** / **Value** Multiplier / 1 Adder / 0
VMAccessAction	UINT16	VM Access Action **Enumeration** **Binary** / **Value** / **Descripion** 0 / ReadSpcl / ReadSpcl 1 / Read / Read 2 / Read1 / Read 3 / Write / Write 4 / Write1 / Write 5 / WriteSpcl / WriteSpcl
VMFunctionCode	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 1 / Read / read 2 / Write / write
VMIndex	UINT16	VM table index for this FeederNetID **Numeric Type, Range: 0-65535** **Action** / **Value** Multiplier / 1 Adder / 0
XferInactiveReason	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 1 / XferDecline / Xfer Decline 2 / XferSuccess / Xfer Success 3 / XferStopped / Xfer Stopped 4 / TimerExpired / Timer Expired
xmtresultcode	UINT16	**Enumeration** **Binary** / **Value** / **Descripion** 0 / NETRETURNSUCCESS / NET return is successful

		1	NETRETURNERROR	NET return has an error

Table 3. Historic Events (1156 items)

Description	Category	Logging Level	Definition
Event Register OK	DAT[MCU]	Normal	The task indicated has enabled event registering for the specifi registering of an event is the process of making all team memb that an event has occurred. Data 1: **Team** Data1 Type: Team Data 2: **Task** Data2 Type: Task Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error Getting Internal 6800 Data	DAT[MCU]	Normal	An error was detected when the IntelliTEAM II software colle related to the internal switch function. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error Getting Local 6800 Data	DAT[MCU]	Normal	An error was detected when the IntelliTEAM II software retrie the local switch in the specified team. This may occur if the sw number configured on the SETUP: Team screen is incorrect. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error Writing Coach Task-List Full	DAT[MCU]	Normal	The list of pending tasks that the coach carries between team n full in the specified team. No more tasks can be put on this list more of the existing tasks have been completed. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error Writing Event Task-List Full	DAT[MCU]	Normal	The list of pending team-related tasks is full in the specified te tasks can be put on this list until one or more of the existing ta completed. Data 1: **Team** Data1 Type: Team

			Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error Writing Member Task-List Full	DAT[MCU]	Normal	The list of pending member-process tasks is full in the specifie more tasks can be put on this list until one or more of the exist have been completed. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error Writing Comm Task-List Full	DAT[MCU]	Normal	The coach or the team member needs to send a new message t team member and the DNP communications buffer is full. Exi transactions must be completed before more are put on the cor list. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error Getting Comm Task From List	DAT[MCU]	Normal	An error was detected when removing a message from the DN communications buffer. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
New Coach Generated	DAT[MCU]	Normal	A new coach has been generated at the local team member for team. This could be caused by a power up state, by the existing lost due to communications failure, or by the existing coach da inconsistent. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Coach Old - Duplicate or CRC Bad	DAT[MCU]	Extended	The coach received by the specified team is not the current coa duplicate of the current coach or it contains data inconsistent v presently expected data. The coach is rejected. Data 1: **Team** Data1 Type: Team Data 2: **Coach Rejection Code** Data2 Type: CoachRejectionC Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Old or Duplicate Task Discarded	DAT[MCU]	All	The task taken from the event list on the specified team is eith duplicate of an existing task. This occurs normally in the oper team as events are distributed throughout the team. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch Not Ready for Transfer	DAT[MCU]	Normal	The local 6800 on the specified team is not ready for transfer This may be caused by an internal switch error (for example, a Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Unknown Event/Task Request	DAT[MCU]	Normal	An event or task for which the local team member is not progr requested within the specified team. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Find Alternate Source Result	DAT[MCU]	Normal	During a transfer event, the team must find an alternate source alternate source sequence and the normal function of the switc the team (both entered on the SETUP: Team screen). This mes indicates the resulting switch record, to be used for the team in Data 1: **Team** Data1 Type: Team Data 2: **Rec** Data2 Type: MemberWithDoneAndEOL Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Register Event Disabled	DAT[MCU]	Normal	The registering of events for distribution within the specified t disabled at the local team member. This was probably caused in the team's configuration on the SETUP: Team screen. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Member Requested	DAT[MCU]	All	The specified team has requested that the local team member e task indicated. Data 1: **Team** Data1 Type: Team Data 2: **Task** Data2 Type: Task

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Close Switch OK	DAT[MCU]	Normal	The switch/position indicated was successfully closed by the s team. Data 1: **Team** Data1 Type: Team Data 2: **Switch Position** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch Closing Issue	DAT[MCU]	Extended	The switch/position indicated failed to close or remain closed by the specified team. Automatic operation may have been dis team member, or the switch may have reopened during the sho time period. Data 1: **Team** Data1 Type: Team Data 2: **Switch position** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transfer Approved	DAT[MCU]	Normal	The transfer operation requested by the specified team, using t source switch previously determined, was approved by the adj The approval code is also shown. For details on the code, cont Data 1: **Team** Data1 Type: Team Data 2: **Restore Condition Test Result** Data2 Type: RestoreConditionTestResult Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transfer Declined	DAT[MCU]	Normal	The transfer operation requested by the specified team has bee the adjacent teams. The requesting team must look for another source, or retry the operation on this alternate source if no othe Possible codes are: (code- definition) 2- Fault isolated, 3- Exc Open team not prepared for the transfer, 5- Closed team not pr the transfer, 6- Line segment limit exceeded, 7- Issue detected teams, 8- Phase loss isolated, 9- Source breaker isolated. Data 1: **Team** Data1 Type: Team Data 2: **Restore Condition Test Result** Data2 Type: RestoreConditionTestResult Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch Open OK	DAT[MCU]	Normal	The specified team successfully opened the switch/position inc Data 1: **Team** Data1 Type: Team Data 2: **Switch Position** Data2 Type: SwitchPosition

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch Opening Unsuccesful	DAT[MCU]	Normal	The specified team was unable to open the switch/position ind Data 1: **Team** Data1 Type: Team Data 2: **Switch Position** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Operation Switch Function Return	DAT[MCU]	Extended	This message displays the internal code returned during the op local switch in the specified team. Contact S&C for code detai is application specific. Data 1: **Team** Data1 Type: Team Data 2: **Switch Operation Result Code** Data2 Type: SwitchOperationResult Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Coach Arrived	DAT[MCU]	All	This message logs the arrival of the coach, along with all the u at the local team member for the specified team. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Coach Has Departed	DAT[MCU]	All	This message logs the departure of the coach from the local tea for the specified team, and indicates where the coach is going Data 1: **Team** Data1 Type: Team Data 2: **Goto Rec** Data2 Type: TeamRecord Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Sequence Numbers Resynchronization	DAT[MCU]	Normal	The sequence numbers of events for the specified team have fa synchronization. The last sequence number received is shown. team member will now resynchronize the number. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Coach Arrived on Request	DAT[MCU]	All	This message logs the arrival of the coach at the local team me specified team after the local team member requested the coac Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Volt/Fault Reset Occurred	DAT[MCU]	Extended	The voltage loss and overcurrent indications maintained by the IntelliTEAM II software have been reset after either the Sectio Time or, if a transfer event has occurred, after team reconfigur complete. Data 1: **Switch Position** Data1 Type: SwitchPosition Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Coach is Held by Team Member	DAT[MCU]	All	The coach for the specified team is being held by the local tea This occurs when a process is taking place at the local team m requires the presence of coaches from both adjacent teams. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Line Segment Faulted	DAT[MCU]	Normal	The line segment protected by the specified team is the locatio overcurrent fault on the circuit. The team will not attempt to re to this line segment. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Device Opened for Transfer	DAT[MCU]	Normal	During a transfer event, the coach of the specified team opene indicated to allow the transfer operation to continue. This may one or more switches within the team are not coordinated to o same time as the other switches. Data 1: **Team** Data1 Type: Team Data 2: **Rec** Data2 Type: SwitchRecord Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Return to Normal Start Event Request	DAT[MCU]	Normal	The Return to Normal timer has expired, allowing the Return t process to start on the specified team. This message indicates t specified team generated the event. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Return To Normal Disabled at Switch	DAT[MCU]	Normal	The Return to Normal process will not be carried out on the sp because Return to Normal is disabled on the SETUP: Team sc Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Return to Normal Timer Started	DAT[MCU]	Normal	The Return to Normal timer was started by the local team mem specified team. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Return to Normal Start Event Received	DAT[MCU]	Normal	The local team member of the specified team received a reque Return to Normal process. This follows the end of the Return t timer and the subsequent event request. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Return to Normal Process Stopped	DAT[MCU]	Extended	The Return to Normal process completed at the local team mer indicated completion code. For details on the code, contact S& Data 1: **Team** Data1 Type: Team Data 2: **Code** Data2 Type: InternalCode Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Return to Normal Continue OK	DAT[MCU]	Normal	The internal Return to Normal process for the specified team i Return to Normal may continue to the next step. The process r also shown. Data 1: **Team** Data1 Type: Team Data 2: **Code** Data2 Type: InternalCode Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Task Travel	DAT[MCU]	All	During a Return to Normal process, tasks associated with the p among multiple teams between the normal source and the norr of the circuit. This message traces the path of the tasks. Data 1: **from Team** Data1 Type: Team Data 2: **Team** Data2 Type: Team Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Return to Normal Process Unsuccessful	DAT[MCU]	Normal	The internal Return to Normal process for the specified team i Return to Normal cannot continue. This may be caused by a te that has Return to Normal disabled (Code 7), or by the adjacer not yet being in its normal state (Code 6). Data 1: **Team** Data1 Type: Team Data 2: **Code** Data2 Type: InternalCode Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Config Update - Operation Suspended	DAT[MCU]	Normal	The team configuration of any of the active local teams is bein the SETUP: Team screen. While this change is in progress, tea is suspended. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Internal Test Point	DAT[MCU]	Normal	A general internal message to display data that may be helpful diagnostics. Please contact S&C if you see this message. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Coach Collect Data	DAT[MCU]	Extended	This message logs the collection of new data by the coach in tl team. This data collection process occurs during the start of a t The team record where the coach is going is also shown. Data 1: **Team** Data1 Type: Team Data 2: **Goto Rec** Data2 Type: MemberWithDoneAndEOL Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Coach Hold Override	DAT[MCU]	All	When the team member of the specified team holds the coach extended period of time, an override occurs that allows the coa visit other team members. This prevents the coach from becon regenerated by an adjacent team member. The team record wh is going is also shown. Data 1: **Team** Data1 Type: Team Data 2: **Goto Rec** Data2 Type: TeamRecord Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Rebuilding Coach	DAT[MCU]	Normal	The coach for the specified team is being regenerated. This ma by a power up event, a configuration change in the team, or a l to communications failure. A diagnostic code is also shown, c

			for code details. Data 1: **Team** Data1 Type: Team Data 2:Task identifier Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Team Not Ready - Discard Task	DAT[MCU]	All	The indicated task has been discarded because the specified tea ready to transfer. This is typically the result of a local or team condition. Data 1: **Team** Data1 Type: Team Data 2: **Task** Data2 Type: Task Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
SCADA Prohibit Restoration Active	DAT[MCU]	Normal	A SCADA command was received to prevent the restoration o this team member (the switch may not close automatically), ho protection is not affected. If applicable, this message will also on power up. Data 1:Debug Data Data1 Type: PRReason Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Timer Prohibit Restoration Active	DAT[MCU]	Normal	The Prohibit Restoration Timer expired, preventing the restora by the team for which the timer expired, however automatic se is not affected. If applicable, this message will also be displaye up. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Closed Loop RSD Center Recalc Event	DAT[MCU]	Normal	Closed Loop RSD Center Recalculation Event. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Extended STL Requested	DAT[MCU]	Normal	Extended STL Requested. Data 1:Debug Data Data1 Type: SW1Or2 Data 2:Debug Data Data2 Type: EnabDisab Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Team Communication Error	DAT[MCU]	Normal	A team-related message could not be delivered. Data 1: **Team** Data1 Type: Team Data 2: **RTU** Data2 Type: RTUAddress Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Monitor Line Segment	DAT[MCU]	All	The specified team has an indication to start a transfer event, b segment is still energized. The team monitors the segment unti deenergized. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Closed Loop Member Center Change	DAT[MCU]	Normal	Closed Loop Member Center Change. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Not All Configured Teams Xfer Ready	DAT[MCU]	Normal	At least one of the active teams where the local control is a me fully operational. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transfer in Progress on Any Team	DAT[MCU]	Normal	A team is in the process of reconfiguring the circuit and transf an alternate source. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RTN in Progress on Any Team	DAT[MCU]	Normal	A team is presently returning the circuit to its normal configur Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Adjust Line Segment Count	DAT[MCU]	Extended	Specified team has increased or decreased the line segment co with the Line Segment Limit set point, following a transfer eve Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Software Mismatch on Arriving Coach	DAT[MCU]	Normal	There is a software revision incompatibility within the team. T show revision and version information for the team member fr coach just arrived. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch Open - Extended Parallel	DAT[MCU]	Normal	During a closed transition Return to Normal, the team member switch automatically opened the switch after a prescribed time insured that a circuit parallel was not left in place indefinitely. condition is not normal, and may have resulted in load being d Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Unexpected State Change	DAT[MCU]	Extended	The transfer state went through an unexpected transition at the team. This error might stop an ongoing transfer process. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Loading Data Reset	DAT[MCU]	Extended	When a transfer with a known load value occurs, the IntelliTE software resets the loading data to reflect the new value. This u information more quickly than the 2-minute load averaging. Tl relates to the state of the reset process. Data 1: **Team** Data1 Type: Team Data 2:Task identifier Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract Receiver Busy	DAT[MCU]	Extended	A contract agent's receiver buffer was full, so a contract messa dropped. The numbers for the specified agent are associated w address at the originating team member. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3: **Agent** Data3 Type: Agent Data 4:Debug Data Data4 Type: ushort
Contract Added to List	DAT[MCU]	All	A new contract was added to the list of contracts being maintai message shows both the requesting and granting teams associa

			transfer process. The requesting team (Team 1) identifies the c contract. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Team Data3 Type: Team Data 4:Team Data4 Type: Team
New Contract Addition Issue	DAT[MCU]	Extended	A contract agent tried to add a new contract to its list but coulc contract was declined. The number for the specified agent is as the RTU address at the originating team member. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract Pending Issue	DAT[MCU]	Extended	The requesting contract agent was waiting for a response wher ran out, so it the contract was unsuccessful. The coach may res contract request if it cannot find another alternate source. This shows the two teams involved with the transfer process at this location, where (Team 1) is the requesting team. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract Request Was Declined	DAT[MCU]	Normal	The granting agent declined the contract request. The number specified agent is associated with the RTU address at the origi member. Note that this message can appear at any agent with t on its list. Data 1: **Agent** Data1 Type: Agent Data 2:SwitchPosition Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract General Error	DAT[MCU]	Extended	The specified contract agent detected a contract error. The nun agent is associated with the RTU address at the team member. Data 1: **Agent** Data1 Type: Agent Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract Request Travel	DAT[MCU]	All	A contract request is traveling between teams. (Team 1) is the through which the request just came, and (Rec 2) is the team n direction where the request is headed.

			Data 1: **Agent** Data1 Type: Agent Data 2: **Team** Data2 Type: Team Data 3: **TeamRecord** Data3 Type: TeamRecord Data 4:Debug Data Data4 Type: ushort
Contract Request Was Accepted	DAT[MCU]	Normal	The granting agent accepted the contract. The number for the s agent is associated with the RTU address at the originating tea Data 1: **Agent** Data1 Type: Agent Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract is Being Dissolved	DAT[MCU]	Extended	An active contract is no longer needed and is in the process of dissolved. The number for the specified agent is associated wit address at the originating team member. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract Started by Member	DAT[MCU]	All	A member of the specified team has determined that it can clos information from the coach, but it must first request a contract Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3: **Team** Data3 Type: Team Data 4: **TeamRecord** Data4 Type: TeamRecord
Contract Approved Switch Close	DAT[MCU]	Normal	The specified team requested a contract, which then traveled to agent, was approved, and came back. The switch has closed to line segment. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3: **Team** Data3 Type: Team Data 4:Debug Data Data4 Type: ushort
Contract Declined	DAT[MCU]	Normal	The contract request made by the specified team was declined. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3: **Team** Data3 Type: Team Data 4:Debug Data Data4 Type: ushort
Contract Requested by Member	DAT[MCU]	Normal	The local team member has requested that the contract agent n contract on behalf of the specified team. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract Requires Member Wait	DAT[MCU]	All	The specified team is waiting for a contract to be requested, gr declined. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3: **Team** Data3 Type: Team Data 4:Debug Data Data4 Type: ushort
Contract Communication Received	DAT[MCU]	All	The contract agent has received a message. (Team 1) refers to originating segment, and (Team 2) refers to the temporary segr the segment through which the message just passed. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4: **ContractState** Data4 Type: ContractState
Contract Maintained	DAT[MCU]	All	Scheduled maintenance of a contract was performed by the spe requesting agent to confirm the continued need for the contrac number for the agent is associated with the RTU address at the member. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Duplicate Contract Received	DAT[MCU]	All	The specified requesting agent received an old or duplicate cor message, which refers to the indicated contract state. The numl agent is associated with the RTU address at the team member. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3: **ContractState** Data3 Type: ContractState Data 4:Debug Data Data4 Type: ushort
Contract Transmit Busy	DAT[MCU]	All	A contract agent's transmit buffer was full. The message is hel transmit buffer has space, and is then sent. The number for the agent is associated with the RTU address at the originating tea Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract Dissolved by Member	DAT[MCU]	Extended	The local team member has started the process to dissolve an a contract on behalf of the specified team.

			Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3: **Team** Data3 Type: Team Data 4:Debug Data Data4 Type: ushort
Contract Resource Limitation	DAT[MCU]	Normal	A contract agent found that resources were not available for lo because of either segment limitations (Code 2) or capacity lim (Code 3). The contract agent did not forward the contract any Data 1: **Agent** Data1 Type: Agent Data 2: **ushort** Data2 Type: ushort Data 3: **ushort** Data3 Type: ushort Data 4: **ushort** Data4 Type: ushort
Contract Cannot Travel	DAT[MCU]	All	The requesting contract agent does not know where the presen it could not forward the contract request. The contract failed. T for the specified agent is associated with the RTU address at th member. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract Not Found	DAT[MCU]	All	A contract agent received a message about a contract that is nc This may result in the contract being dissolved and, if necessar reactivated. The number for the specified agent is associated w address at the originating team member. Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Contract Reactivated	DAT[MCU]	Extended	The contract is missing somewhere along its routing path, so th agent reactivated the contract. The number for the specified ag associated with the RTU address at the originating team memb Data 1: **Agent** Data1 Type: Agent Data 2: **SwitchPosition** Data2 Type: SwitchPosition Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Alternate Source Flag Cleared	DAT[MCU]	Extended	The line segment associated with the specified team is no long from an alternate source. This message usually follows a Retu operation. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Member Cleared Task Lock Attributes	DAT[MCU]	All	The team member logic cleared the execution lock on tasks pr task list. These tasks may now be executed by the team memb Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Pending Comm Message Cleared	DAT[MCU]	All	The coach has determined that a pending message is no longer should be removed from the communications transmit list. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Prohibit Restoration Timer Expired	DAT[MCU]	Normal	The timer for the Prohibit Restoration feature has expired and Prohibit Restoration to become active. (Team 1) indicates the t to which this event applies. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Action Path Complete	DAT[MCU]	All	This message is displayed when the action path for operating t gear has completed all possible steps in either the forward or r direction. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Next Action	DAT[MCU]	All	The operation of the switch gear is progressing to the next acti action path. (Action 1) indicates the action to be taken and can following codes: 3 Close for xfer, 20 Contract request, 21 Cor Terminate, 30 Block recloser, 31 Unblock recloser, 33 Block g 34 Unblock ground trip, 36 Alternate settings, 37 Normal, 253 done. (Direction 2) is the direction the action path is going, ei (1) or Reverse (2). Data 1: **Action** Data1 Type: Action Data 2: **Direction** Data2 Type: Direction Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Control Feature OK	DAT[MCU]	Extended	Indicates the requested control feature executed normally. Pos values are: 1 Point to operate switch, 2 Point to block reclosin block ground trip, 4 Point to change profile. Data 1: **Team** Data1 Type: Team Data 2: **Point** Data2 Type: Point Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Control Feature Unsuccessful	DAT[MCU]	Normal	Indicates the requested control feature did not execute normall Points: 1= Operate 6800, 2= Block reclosing, 3= Block ground Change profile. Data 1: **Team** Data1 Type: Team Data 2: **Point** Data2 Type: Point Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Alternate Source Flag Set	DAT[MCU]	Normal	Alternate Source Flag Set. Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Closed Loop RSD Recalc Timer Expiration	DAT[MCU]	Normal	Closed Loop RSD Recalculation Timer Expiration. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Closed Loop Member Revert Center	DAT[MCU]	Normal	Closed Loop Member Revert Center. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Closed Loop Incorrect Circuit Topology	DAT[MCU]	Normal	Closed Loop Incorrect Circuit Topology. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Volt/Fault Idle Transfer State	DAT[MCU]	All	This message is output when all teams that the 6800 is a memb their transfer stated back to idle, signaling a reset of the total 3 average load. Data 1: **Switch Position** Data1 Type: SwitchPosition

			Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Closed Loop Incorrect RT Load	DAT[MCU]	Normal	Closed Loop Incorrect RT Load. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Volt/Fault Overcurrent Cleared	DAT[MCU]	All	This message is output when an overcurrent fault was previous the field now is not faulted and 3 phase voltage has returned, c coach to clear the latched overcurrent condition. For details on contact S&C. Data 1: **Team** Data1 Type: TeamRecord Data 2: **Rec** Data2 Type: SwitchRecord Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Volt/Fault Voltage Loss Cleared	DAT[MCU]	All	The coach clears a 3 phase voltage loss when: 3 phase voltage previously detected and real time 3 phase voltage is now prese the external device is in its normal state or the external device' is a source sub. For details on Rec 2, contact S&C. Data 1: **Team** Data1 Type: TeamRecord Data 2: **Rec** Data2 Type: SwitchRecord Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Volt/Fault Phase Loss Cleared	DAT[MCU]	All	This message is output when a phase loss was previously dete external device is now in its normal open or close state, real ti voltage is present, causing the coach to clear the latched phase condition. For details on (Rec 2) contact S&C. Data 1: **Team** Data1 Type: TeamRecord Data 2: **Rec** Data2 Type: SwitchRecord Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Feeder Loading Data Received	DAT[MCU]	All	Feeder loading data has been received from the source substati breaker, and may be used in determining the capacity of the ci transfer operations. Data 1 indicates the circuit loading receive increments of 10 amps per count. Data 1: **Data** Data1 Type: LoadingCurrent Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Error Cleared - Gathering Data	DAT[MCU]	Normal	The error collecting data related to the internal switch function Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Device is Ready for Transfer	DAT[MCU]	Normal	The local switch on the specified team is ready for transfer ope Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Config Update Operation Resumed	DAT[MCU]	Normal	The team configuration of any of the active local teams has be on the SETUP: Team screen. Team operation is resumed. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
SCADA Prohibit Restoration Cleared	DAT[MCU]	Normal	A SCADA command was received to reenable the restoration this team member, and the switch may be closed automatically applicable, this message will also be displayed on power up. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Timer Prohibit Restoration Cleared	DAT[MCU]	Normal	A SCADA command was received to reenable the restoration this team member, and the switch may be closed automatically applicable, this message will also be displayed on power up. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
All Teams Are Transfer Ready	EVT[MCU]	Normal	All teams are fully operational, and may close switches as nec transfer load and reconfigure the circuit. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transfer Not Active on Any Team	DAT[MCU]	Normal	No teams are presently reconfiguring the circuit or transferring Data 1:Debug Data Data1 Type: ushort

			Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RTN Not Active on Any Team	DAT[MCU]	Normal	No teams are presently returning the circuit to its normal confi Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Unknown Message Type Received	DAT[MCU]	Normal	An IntelliTEAM message was received over communications message type that is not recognized. The team the message wa for, and the message type received, are included in the data. Data 1: **Team** Data1 Type: Team Data 2: **Message Type Received** Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Request Xfer Trip Prohibit Restoration	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Req Xfer Trip Prohibit Rest on DG Team	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Xfer Trip Prohibit Restoration Active	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Xfer Trip Prohibit Restoration Cleared	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DG Reconnect Delay Started	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DG Reconnect Delay Expired	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Not All Prohibit Rest Flags Cleared	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Request Switch Close For DG Reconnect	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch Closed For DG Reconnection	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DG Reconnect Delay Terminated	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DG Reconnect Delay Disabled	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DG Reconnect Delay Counting Down	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Start Monitoring TTPR	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Continue Monitoring TTPR	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Could Not Find DG	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Could Not Find Normal Source	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Stop Monitoring TTPR	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Source Side Team Not Defined	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Could Not Start Monitoring TTPR	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Closed Loop Center Determination Problem	DAT[MCU]	Normal	Closed Loop Center Determination Problem. Data 1:Debug Data Data1 Type: ushort

			Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transfer Declined	DAT[MCU]	Normal	Data 1: **Team** Data1 Type: Team Data 2: **ExcessLoad** Data2 Type: ExcessLoad Data 3: **Unused** Data3 Type: ushort Data 4: **Debug Data** Data4 Type: HealingType
Transfer Declined Excess Load Inactive	DAT[MCU]	Normal	Data 1: **Team** Data1 Type: Team Data 2: **Debug Data** Data2 Type: XferInactiveReason Data 3: **Unused** Data3 Type: ushort Data 4: **Unused** Data4 Type: ushort
Transfer Declined	DAT[MCU]	Normal	Data 1: **Team** Data1 Type: Team Data 2: **LineSegLimit** Data2 Type: LineSegLimit Data 3: **Unused** Data3 Type: ushort Data 4: **Debug Data** Data4 Type: HealingType
Transfer Declined Line Seg Limit Off	DAT[MCU]	Normal	Data 1: **Team** Data1 Type: Team Data 2: **Debug Data** Data2 Type: XferInactiveReason Data 3: **Unused** Data3 Type: ushort Data 4: **Unused** Data4 Type: ushort
DG Reconnect Delay Terminated Cleared	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Normally Open Switch Automatic Open	DAT[MCU]	Normal	A normally open team member has opened for an automatic (I initiated) reason. This may happen during Return To Normal p team and the member record number are provided. Data 1: **Team** Data1 Type: Team Data 2: **Rec** Data2 Type: SwitchRecord Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Message Rejected	DAT[MCU]	Normal	An IntelliTEAM communication message to the provided RTU rejected by communications. The rejection code is provided. Data 1: **Rtu** Data1 Type: RTUAddress

			Data 2: **Code** Data2 Type: CommErrorCode Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transfer State Change	DAT[MCU]	All	The transfer state changed for the specified team. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Manual Operation Team Condition On	DAT[MCU]	Normal	A team entered a non-operational state because an unexpected switch operation occurred. Typically the only expected manua operation is closing a source switch on a previously faulted tea proves that the fault is gone, and allows RTN process to take p is enabled). Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Manual Operation Team Condition OFF	DAT[MCU]	Normal	The unexpected manual switch operation team condition was c can only occur as a result of a user request. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Manual Operation Not Cleared	DAT[MCU]	Normal	Manual Operation condition could not be cleared on a user req the local team member is not in its normal state. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Wait For Team to Open	DAT[MCU]	Normal	IntelliTEAM is waiting for all switches in the team to open so attempt to restore service to the team. This is likely to occur or 2nd contingency event. Data 1: **team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Bad Voltage Reopen Flag	DAT[MCU]	Normal	Associated with the Team Member Requalify Time feature. Va means that the team member is disqualified on an unsuccessfu close and a timer is started. Value of 0 means that the timer ha the team member can be considered as an alternate source agai

			Data 1: **Team** Data1 Type: Team Data 2: **Flag value** Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
TII Source Loading Data Active	DAT[MCU]	Normal	
TII Source Loading Data Not Active	DAT[MCU]	Normal	
TII RT-Load Data Problem	DAT[MCU]	Normal	
TII RT-Load Data Problem Cleared	DAT[MCU]	Normal	
TII Cycling Status Cleared(team rec)	DAT[MCU]	Normal	Data 1: **Team** Data1 Type: TeamRecord Data 2: **Rec** Data2 Type: SwitchRecord
Team 1 is Ready to Transfer	EVT[MCU]	Extended	
Team 1 is in Not Ready state	EVT[MCU]	Extended	
Team 2 is Ready to Transfer	EVT[MCU]	Extended	
Team 2 is in Not Ready state	EVT[MCU]	Extended	
Team 3 is Ready to Transfer	EVT[MCU]	Extended	
Team 3 is in Not Ready state	EVT[MCU]	Extended	
Team 4 is Ready to Transfer	EVT[MCU]	Extended	
Team 4 is in Not Ready state	EVT[MCU]	Extended	
Team 5 is Ready to Transfer	EVT[MCU]	Extended	
Team 5 is in Not Ready state	EVT[MCU]	Extended	
Team 6 is Ready to Transfer	EVT[MCU]	Extended	
Team 6 is in Not Ready state	EVT[MCU]	Extended	
Team 7 is Ready to Transfer	EVT[MCU]	Extended	
Team 7 is in Not Ready state	EVT[MCU]	Extended	
Team 8 is Ready to Transfer	EVT[MCU]	Extended	
Team 8 is in Not Ready state	EVT[MCU]	Extended	
RSH start after switch open	DAT[MCU]	All	Rapid Self Healing is being started after a switch has opened t trouble. Data 1: **Team** Data1 Type: ushort
RSH failed - downstream team	DAT[MCU]	All	Rapid Self Healing was unable to restore load based on inform team on the load side of the fault. Data 1: **Team** Data1 Type: ushort
RSH failed - faulted team	DAT[MCU]	All	Rapid Self Healing was unable to restore load based on the inf from the faulted team. Data 1: **Team** Data1 Type: ushort

RSH monitoring in process	DAT[MCU]	All	Indicates that a coach monitoring task was started to watch the the Rapid Self Healing process. This task will be used to rever standard restoration process if Rapid Self Healing is unable to load. Data 1: **Team** Data1 Type: ushort
RSH has become active	DAT[MCU]	All	Logged when the coach is requested to make Rapid Self Heali that team. Data 1: **Team** Data1 Type: ushort
RSH has become inactive	DAT[MCU]	All	Logged when the coach is requested to make Rapid Self Heali that team. Data 1: **Team** Data1 Type: ushort
RSH is being started by player	DAT[MCU]	All	Logged when the coach has requested the team member to init Rapid Self Healing process. Data1= Team Number, Data2= RS Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH failed - reported by player	DAT[MCU]	All	If the team member finds that Rapid Self Healing is unable to will log this message. If there are other tie points that may be u it will try those, indicated tby the Try Number. The result code of: (if code location 0 or 2) 1= No Alternate Source, 2= Not En Capacity, 3= Switch Trouble, 4= Timer Unavailable, 5= Unabl Peer, 6= Comm List Full, 7= Bad Try Number, 8= Bad Netlist Enabled. (if code location 1) 3=Problem while waiting for RSI Timeout while waiting for RSH. Data1= Team Number, Data2 Code, Data3= Try Number, Data4= Code Location. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH action requested by player	DAT[MCU]	All	The member's initial request to start Rapid Self Healing was su Data1= Team Number. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH action successful	DAT[MCU]	All	The team member has received an indication that Rapid Self H been successful. Success code will be 2. Data1= Team Numbe Success Code.

			Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH action taking place	DAT[MCU]	All	The team member has initiated a Rapid Self Healing operation waiting for a result. Data1= Team Number. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH init at startup	DAT[MCU]	All	A peer device was unable to be registered with the DNP comm system. The return code can be: 1= Error, 2= Already On List, 4= Not On List, 5= Not Peer-To-P{eer Compatible, 6= Delay Retrying. Data1= Return Code, Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH timer was not available	DAT[MCU]	Normal	Logged during startup if no timer is available for the Rapid Se process. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH disabled or switch trouble	DAT[MCU]	All	If the switch is in manual mode, or IntelliTEAM event process been in the associated teams, this message will be logged and Healing will not be allowed. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH trouble registering DNP peer	DAT[MCU]	All	A peer device was unable to be registered with the DNP comm system. The return code can be: 1= Error, 2= Already On List, 4= Not On List, 5= Not Peer-to-Peer Compatible, 6= Delay Be Retrying. Data1= Return Code, Data2= Peer Address, Data4= Location. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH throw back operation	DAT[MCU]	All	This indicates that the tie switch has completed its Rapid Self operation and sending a response message back to the originat The status can be: 6= Success, 7= Unable to Close Switch. Da Switch Address, Data2= Originating Switch Address, Data3= Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH peer removed after complete	DAT[MCU]	All	The reported switch address has been removed from the DNP communications peer association list. Data1= Switch address. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH trouble starting operation	DAT[MCU]	All	Rapid Self Healing was unable to begin due to one of the follo 1= No Alternate Source, 2= Not Enough Capacity, 3= Switch Timer Unavailable, 5= Unable to Register Peer, 6= Comm Lis Try Number, 8= Bad Netlist, 9= Not RSH Enabled. Data1= Tr Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH initiated	DAT[MCU]	All	Logged at the originating team member when Rapid Self Heal initiated successfully. The message sent to the tie point includ of teams that will be restored, and the amount of load that will if the RSH action is executed at the tie switch. A response is th to find out whether the tie switch was able to close. Data1= Ti Address, Data2= Tie Switch Position, Data3= Number of Tear Load to Restore. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH switch operation trouble	DAT[MCU]	All	Logged if the tie switch location is unable to close the switch. reason for this trouble is related to the code location: 1= Rapid Inactive, 2= Request Team Invalid, 3+4= RSH Disabled on Te Team Errors Present, 7= Switch Is Not Open, 8= Voltage State Team Is Faulted, 10= Too Many Teams to Restore, 11+12= N Capacity, 13= Coach Task List Full, 14= Grant Team Invalid.

			for further definitions of Data1, Data2, and Data3. Data4= Coc Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH switch op final status	DAT[MCU]	Normal	Indicates the status of operating a switch based on Rapid Self H Switch operation status includes: 1= Good Operation, 2= Bad Operation Timed Out. The RSH process status can be: 4= Tim Good Status, 7= Bad Status. Data1= Switch Operation Status, Process Status, Data4= Code Location. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH rcvd unexpected state change	DAT[MCU]	All	Logged if a Rapid Self Healing message is received that conta that conflicts with the present local RSH state. In general, if R then a request to close may be accepted, and if a request to clo transmitted then only a result response message can be accepte states include: 1= Idce, 2= Waiting, 3= Error, 4= Timeout, 5= 6= Result Response, 7= Remove Peer, 8= Waiting, 9= Respon Present RSH State, Data2= Received RSH State, Data3= Prese Status. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Netlist initialization trouble	DAT[MCU]	Normal	If a Netlist is unable to be imported into the Team database thi logged to indicate the reason. Data1= Trouble Code (1= Prese local switches do not match normal states, 2= IntelliTEAM ma process of reconfiguration, 3= Netlist data is invalid, such as n 4= Presently processing a new netlist, 5= Device is not configu RTU address). Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Netlist accepted by IntelliTEAM	DAT[MCU]	Normal	Indicates that a Netlist was successfully imported into the Tea The count of the number of teams included is also reported. Da Count. Data 1:Debug Data Data1 Type: ushort

			Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Netlist rejected by IntelliTEAM	DAT[MCU]	Normal	During import into the team database the Netlist was rejected e this device not being included in the Netlist, or trouble with th unpacking the Netlist. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RSH alt source search result	DAT[MCU]	Extended	Logged when the Rapid Self Healing process searches for alter during a reconfiguration process. The number of possible alter Found is reported, along with the number Available for restora checking each alternate source for available capacity. Data1= I Data2= Available. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Loading Agent Timer Unavailable	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load Shedding Initiated	DAT[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Trouble Starting Load Shedding Operation	DAT[MCU]	Normal	A new coach has been generated at a team member of the spec This can occur during power-up, if existing coach is lost due to communication failure, or existing coach data is inconsistent. I Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Recvd Shed Rqst While in Process	DAT[MCU]	Extended	Indicates that a request to shed load was received while a previ is in process. Data1= State (1= Start, 2= Return Response, 3= (Remove Peer, 5= Back to Normal, 65535= Idle), Data2= Targ

			Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Trouble During Shed Operation	DAT[MCU]	Extended	Indication of trouble during shed operations is dependent on th Location. Code Location 1 indicates the target switch is not pr closed (Data1 indicates switch state). Code Location 2 indicate team numbers are inconsistent (Data1 and Data2 are actual tea Data3 is the requested team number). Code Location 3 indicat errors may be present (Data1 and Data2 are associated team nu Code Location 6 indicates that the priority of the load to be sh to shed (Data1 shows the configured load priority). Code Loca indicates that there was trouble returning a load shed response (Data1 is the destination address). Data1, Datas2, and Data3 se Data4= Code Location. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load Agent Throwback Operation	DAT[MCU]	All	A load shed response message was sent back to the requesting destination address is included. Data1= Destination. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load Agent Trouble Adding DNP Peer	DAT[MCU]	Extended	This is logged when a destination device for a load shed messa to be added (code location 2) or removed (code location 1) fro peer list Data1= Return Code (1= Error, 2= Already on list, 3= not on list) Data2= Destination, Data4= Code Location. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load agent removed DNP peer	DAT[MCU]	All	A destination peer device was successfully removed from the l after a load shed operation was completed. This prevents the p filling up over time. Data1= Destination. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Load agent remove restrictions	DAT[MCU]	Extended	Indicates that a switch that was used to shed load can again be source for that load. The restriction from closing has been rem Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load Shedding Switch Mode Status	DAT[MCU]	All	This is logged to show whether the Switch Mode status indicat team member is being blocked from operations due to a load s operation. Data1= Team Number, Data2= Member Number, D (1= Transition to On, 2= Transition to Off). Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Overload monitoring started	DAT[MCU]	Normal	Indicates beginning of load monitoring for the substation asso this control since it is supplying sections that it normally does Data1= Source List Index in Net View. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Overload monitoring stopped	DAT[MCU]	Normal	Indicates that load monitoring is no longer required for this so is no longer carrying non-normal sections. Data1= Source List View, Data2= Source Capacity, Data3= Present Load. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Qualified Overload is present	DAT[MCU]	All	Indicates that an overload condition has been present for the sp length of time - this is the declaration we are not seeing a mom transient. Data1= Source List Index in Netview, Data2= Sourc Data3= Present Load. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Overload Alarm Active	DAT[MCU]	Normal	Alarm is active once an overload has been declared. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Overload Alarm Removed	DAT[MCU]	Normal	Overloadondition has cleared and/or the source is no longer ca sections. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load shed sequence started	DAT[MCU]	Extended	Indicates that the time on alarm state limit has passed, and an a be made for a section to be transferred or shed to attempt to all condition. Data1= Source List Index in Net View, Data2= Sou Data3= Present Load. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load shed request sent	DAT[MCU]	All	Shed request was sent to the specified RTU address. Data1= D Switch RTU, Data2= Destination Switch Team Number, Data Request. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load shed request not sent	DAT[MCU]	All	Problem with communications prevented sending a shed reque specified RTU address. Data1= Destination Switch RTU, Data Destination Switch Team Number, Data3= Load Request. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load shed request accepted	DAT[MCU]	All	Target RTU address device accepted the request and will act o Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Load shed request rejected	DAT[MCU]	All	Target RTU address device rejected the request and will not ac Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Add to shed reject list	DAT[MCU]	All	Target RTU address will not be asked again during this overlo occurrence. Data1= Destination RTU address Data2= Quantity Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Calculated alt source capacity	DAT[MCU]	All	This message indicates the resources that are available on an a feeder as calculated during the RSH process. Data1= Return C Nodata, 1= Netlist update occurring, 2= Netlist good), Data2= Data3= Segment Count, Data4= Netlist Team ID. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Close Fail - Pulsing In-Op	DAT[MCU]	All	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch Open Request	DAT[MCU]	Normal	Target RTU address will not be asked again during this overlo occurrence. Data 1: **Team Number** Data1 Type: Team Data 2:Player Task ID Data2 Type: ushort Data 3: **Switch Position Number** Data3 Type: SwitchPosition Data 4:Debug Data Data4 Type: ushort
Team Dropped for Load Shed	DAT[MCU]	Normal	The team has been deenergized to reduce substation loading ar restored until the circuit is back to normal. Data 1:Debug Data Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
TPR due to Load Shed Active	DAT[MCU]	Normal	Prohibit Restoration is active to prevent shed load from being r an alternate source. Data 1:Debug Data Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

PR due to Load Shed InActive	DAT[MCU]	Normal	Prohibit Restoration is inactive to prevent shed load from bein from an alternate source. Data 1:Debug Data Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch Open For Phase Isolation ON	DAT[MCU]	Normal	Switch Opened For Phase Isolation Active Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Switch Open For Phase Isolation OFF	DAT[MCU]	Normal	Switch Opened For Phase Isolation NOT Active Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Phase Loss Timing Complete	DAT[MCU]	Normal	Phase Loss Timing Complete Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Phase Loss Isolation Event Continue	DAT[MCU]	Normal	Phase Loss Isolation Event Continue Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Sw1 Auto Manl Op. Clear Timer Started	DAT[MCU]	Normal	Sw1 Auto Manual Operation Clearing Timer Started Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Xfer Declined Load Data	DAT[MCU]	Normal	Xfer Declined Load Data Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Load Shed Alarm Active	DAT[MCU]	Normal	Alarm is active once arequirement to shed load been declared.

			Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Sys Voltage Classification Diff Detected	DAT[MCU]	Normal	Alarm is active once arequirement to shed load been declared. Data 1:Debug Data Data1 Type: SystemVoltage Data 2:Debug Data Data2 Type: SystemVoltage Data 3:Debug Data Data3 Type: MoreThanOneDiffFound Data 4:Debug Data Data4 Type: ushort
Load Shed Alarm Removed	DAT[MCU]	Normal	No longer need to shed load - load is below the limit or back t Feeder Configuration. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Return to Loop Timer Started	DAT[MCU]	Normal	The Return to Loop timer was started by the local team memb specified team. Data 1: **Team** Data1 Type: Team Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Return to Loop Continue OK	DAT[MCU]	Normal	The internal Return to Loop process for the specified team ind Return to Loop may continue to the next step. The process res also shown. Data 1: **Team** Data1 Type: Team Data 2: **Code** Data2 Type: InternalCode Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Disregard First Overcurrent Enabled	DAT[MCU]	Normal	Disregard First Overcurrent Enabled. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Disregard First Overcurrent Disabled	DAT[MCU]	Normal	Disregard First Overcurrent Enabled. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

OC Indication Requires Clearing	DAT[MCU]	Normal	OC Indication Requires Clearing Data 1:Debug Data Data1 Type: TeamNumber Data 2:Debug Data Data2 Type: SWPos Data 3:Debug Data Data3 Type: TransferState Data 4:Debug Data Data4 Type: ushort
gnoring First OC is Active	DAT[MCU]	Normal	Ignoring first OC is active Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
gnoring First OC is Inactive	DAT[MCU]	Normal	Ignoring First OC is Inactive Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Average Loading Stopped	DAT[MCU]	Normal	Average Loading Stopped Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Average Loading Restored	DAT[MCU]	Normal	Average Loading Restored Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Calculated Real Capacity	DAT[MCU]	Normal	Calculated Real Capacity Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Not All Teams Xfer Rdy Timer Active	DAT[MCU]	Normal	Not All Teams Xfer Rdy Timer Active Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Not All Teams Xfer Rdy Timer Cleared	DAT[MCU]	Normal	Not All Teams Xfer Rdy Timer Cleared Data 1:Debug Data Data1 Type: ushort

			Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Start Sending Transfer Trip	DAT[MCU]	Normal	Start Sending Transfer Trip Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transfer Trip Sending Rejected by RTL	DAT[MCU]	Normal	Transfer Trip Sending Rejected by RTL Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transfer Trip Sent Successfully	DAT[MCU]	Normal	Transfer Trip Sent Successfully Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transfer Trip Sent Unsuccessfully	DAT[MCU]	Normal	Transfer Trip Sent Unsuccessfully Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Compact Flash Operational Issue	LOG[MCU]	All	Data 1:Error Code Data1 Type: ushort
Logging Overflow (last events)	LOG[MCU]	All	Data 1: **Event ID** Data1 Type: ushort Data 2: **Event ID** Data2 Type: ushort Data 3: **Event ID** Data3 Type: ushort Data 4: **Event ID** Data4 Type: ushort
DNP Irregularity	LOG[MCU]	All	Data 1:Trace Info Data1 Type: ushort Data 2:Error Code Data2 Type: ushort
High Volume Event Storage to CF	LOG[MCU]	All	The task that writes the compact flash file has been reschedule immediately because the event log input buffer is full.

Invalid Log Request	LOG[MCU]	All	Data 1:Trace Info Data1 Type: ushort
Diag. Data Definition Error	LOG[MCU]	Normal	Data 1: **Diagnostic Data Type (Alarm or Warning or Error** LOGDiagType Data 2:On-Event handle Data2 Type: ushort Data 3:Off-Event handle Data3 Type: ushort Data 4:Trace Info Data4 Type: ushort
Diag. Data Processing Error	LOG[MCU]	Normal	Data 1: **Diagnostic Data Type (Alarm or Warning or Error** LOGDiagType Data 2:Debug Data2 Type: ushort Data 3:Debug Data3 Type: ushort Data 4:Trace Info Data4 Type: ushort
Alarm Condition is ON	LOG[MCU]	Normal	
Alarm Condition is OFF	LOG[MCU]	Normal	
Warning Condition is ON	LOG[MCU]	Normal	
Warning Condition is OFF	LOG[MCU]	Normal	
Error Condition is On	LOG[MCU]	Normal	
Error Condition is Off	LOG[MCU]	Normal	
Next Averaging Period	LOG[MCU]	All	Data 1:Period Number Data1 Type: ushort
Averaging Data CF Write	LOG[MCU]	All	Data 1:File Cycle Number Data1 Type: ushort
Next Daily High/Low Period	LOG[MCU]	All	Data 1:Period Number Data1 Type: ushort
Daily High/Low CF Write	LOG[MCU]	All	Data 1:File Cycle Number Data1 Type: ushort
Shutdown Processing LOG	LOG[MCU]	Normal	Data 1:Trace Info Data1 Type: ushort
Communication Processor Startup	EVT[MCU]	Normal	
Spec. Evt Counters Cleared	LOG[MCU]	Normal	
Status Counters Cleared	LOG[MCU]	Normal	

Log Flooding Condition ON	LOG[MCU]	Normal	Data 1:event ID Data1 Type: ushort
Log Flooding Condition OFF	LOG[MCU]	Normal	
CF Query Action	LOG[MCU]	All	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
CF Query Received	LOG[MCU]	All	Data 1: **Client-generated query code** Data1 Type: ushort
CF Query Frame Furnished	LOG[MCU]	All	Data 1: **Next Frame Sequence Number** Data1 Type: ushort Data 2: **Position in CF read buffer** Data2 Type: ushort Data 3: **Number of events in this frame** Data3 Type: ushort
CF Query New Frame Rqst	LOG[MCU]	All	Data 1: **Frame Sequence Number** Data1 Type: ushort
CF Query: Last Frame Furnished	LOG[MCU]	All	Data 1: **Next Frame Sequence Number** Data1 Type: ushort Data 2: **Position in CF read buffer** Data2 Type: ushort Data 3: **Number of events in this frame** Data3 Type: ushort
CF Query Abort Request	LOG[MCU]	All	Data 1: **Frame Sequence Number** Data1 Type: ushort
CF Query Has Timed Out	LOG[MCU]	All	
CF Query Disk Issue	LOG[MCU]	All	Data 1: **Issue Code** Data1 Type: ushort Data 2: Data2 Type: ushort
User Configuration Backed Up to BMM	SUM[MCU]	Normal	User configurable settings successfully stored to the base mem
BMM Setting Restoration Unsuccessful	SUM[MCU]	Normal	An issue was encountered restoring user settings from the base The settings were not restored. Data 1:SUM Error Code Data1 Type: ushort Data 2:Called Function Return Data2 Type: ushort Data 3:Step Data3 Type: ushort
User Configuration Restored From BMM	SUM[MCU]	Normal	User configurable settings successfully restored from the base

BMM Backup Configuration Unsuccessful	SUM[MCU]	Normal	An issue was encountered storing user settings to the base mer settings were not succesfully stored. Data 1:SUM Error Code Data1 Type: ushort Data 2:Called Function Return Data2 Type: ushort Data 3:Step Data3 Type: ushort
User Configuration Backed Up to CF	SUM[MCU]	Normal	User configurable settings successfully stored to a compact fla
CF Backup Settings Save Unsuccessful	SUM[MCU]	Normal	An issue was encountered storing user settings to a compact fl settings were not succesfully stored. Data 1:SUM Error Code Data1 Type: ushort Data 2:Called Function Return Data2 Type: ushort Data 3:Step Data3 Type: ushort
User Configuration Restored to CF	SUM[MCU]	Normal	User configurable settings successfully restored from the a cor file.
CF Restoration Settings Unsuccessful	SUM[MCU]	Normal	An issue was encountered restoring user settings from a compa The settings were not restored. Data 1:SUM Error Code Data1 Type: ushort Data 2:Called Function Return Data2 Type: ushort Data 3:Step Data3 Type: ushort
Validate Settings Successful	SUM[MCU]	Normal	The settings are valid.
Settings Validation Unsuccessful	SUM[MCU]	Normal	An issue was encountered during a setting validation attempt. are not valid and cannot be applied. Data 1: **Settings Group** Data1 Type: ushort Data 2: **Instance** Data2 Type: ushort Data 3: **Error** Data3 Type: ushort
Applied Settings Successfully	EVT[MCU]	Normal	The settings were applied and are active in the control.
Settings Application Unsuccessful	EVT[MCU]	Normal	An issue was encountered while trying to apply settings to the settings were not successfully applied. This most commonly o settings are transferred from the MCU to the DSP. Data 1:Error Code Data1 Type: ushort
Refresh Settings Buffer Successfully	SUM[MCU]	Normal	The settings buffer has been refreshed with the active settings.
Refresh Settings Buffer Issue	SUM[MCU]	Normal	An issue was encountered while attempting to refresh the setti Values on the settings screens may not accurately reflect the a in the control. Data 1:Error Code Data1 Type: ushort
Register Settings Block Issue	SUM[MCU]	Normal	An issue was encountered during application initialization that group of settings from being successfully registered with the s Data 1:Error Code Data1 Type: ushort

Register Callback Issue	SUM[MCU]	Normal	An issue was encountered during application initialization that settings related functions from a subsystem from being succes registered with the setup manager. Data 1:Error Code Data1 Type: ushort
Register DNP Special Function Warning	SUM[MCU]	Normal	An issue was encountered during application initialization that the functionality of one or more DNP commands. Data 1:DNP Error Code Data1 Type: ushort Data 2:Instance Data2 Type: ushort
Register Non-Settings Block Issue	SUM[MCU]	Normal	An issue was encountered during application initialization that group of non-settings from being successfully registered with t manager for access rights checking. Data 1:Error Code Data1 Type: ushort
User Login Access Denied	SUM[MCU]	Normal	Access to Login was denied. Data 1: **Access Denied Reason** Data1 Type: ushort
User Logout Access Denied	SUM[MCU]	Normal	Access to Logout was denied. Data 1: **Access Denied Reason** Data1 Type: ushort
BMM Not Ready	SUM[MCU]	Normal	The base memory module cannot be accessed. Data 1:BMM State Data1 Type: ushort
User Session Started	EVT[MCU]	Normal	A user session was started. Data 1: **User Id** Data1 Type: ushort Data 2: **Port Number** Data2 Type: PortCode
User Session Ended	EVT[MCU]	Normal	A user session was terminated. Data 1: **User Id** Data1 Type: ushort Data 2: **Session Ended Reason** Data2 Type: SUMSessionEnd Data 3: **Port Number** Data3 Type: PortCode
User Access Denied	SUM[MCU]	Normal	Access to write a VM location was denied. Data 1: **Access Denied Reason** Data1 Type: ushort Data 2:Optional Data Depending on Reason Data2 Type: usho
Register Commad to Block Issue	SUM[MCU]	Normal	An issue was encountered during application initialization that command from being successfully registered with the setup ma block based on source (IntelliLink WiFi or IntelliLink Remote Data 1:Error Code Data1 Type: ushort
Request BMM Backup Received	SUM[MCU]	Normal	The Setup Manager received a request from another subsysten backup to base memory.

Apply Settings Started	SUM[MCU]	Normal	Apply of user configurable settings has started.
Password Setting Changed	SUM[MCU]	Normal	A user group password settings change was applied. Data 1: **GroupId** Data1 Type: ushort
IntelliLink Intrusion On	SUM[MCU]	Normal	User attempted three unsuccessful logons.
IntelliLink Intrusion Off	SUM[MCU]	Normal	Alarm re user attempting three unsuccessful logons was cleare
IntelliLink Intrusion Attempt	SUM[MCU]	Normal	User attempted unsuccessful logon
WFC Transfer to CF Successful	EVT[MCU]	Normal	Data 1: Data1 Type: CycleNumber Data 2: Data2 Type: SecondaryEvent Data 3: Data3 Type: NumberFrames
DNP Initialization Complete	DNP[MCU]	Normal	Displayed when initialization of DNP processes has completec error. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Route Table Initialization Error	DNP[MCU]	Normal	The routing table was unable to be initialized, preventing initia DNP processes from being completed. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Point Map Initialization Error	DNP[MCU]	Normal	The point mapping table was unable to be initialized, preventi initialization of DNP processes from being completed. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error in Configuration Data	DNP[MCU]	Normal	A configuration setup parameter was found to be set incorrectl range. Data 1: **Setup Value** Data1 Type: ushort Data 2: **Min Value** Data2 Type: ushort Data 3: **Max Value** Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Old Configuration Remains in Effect	DNP[MCU]	Normal	Indicates that due to an error in configuration data the existing continue to be used by the application. Data 1:Debug Data Data1 Type: ushort

			Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Configuration Change Accepted	DNP[MCU]	Normal	Changes to the communications setup parameters were validat been made active within the application. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Master Record Not Added	DNP[MCU]	Normal	The master record was unable to be added to the peer device li either an incorrect parameter or the peer device list is full. Data 1: **Master Record Add Error** Data1 Type: MasterRecA Data 2: Code Location Data2 Type: ushort Data 3:Master Number Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Map Change Callback Init Error	DNP[MCU]	Normal	A DNP point map change callback function was unable to be r to a full callback list. This prevented initialization of DNP pro being completed. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Incorrect Frame Length	DNP[MCU]	Extended	The actual length of the received DNP frame does not match tl indicated in the DNP data link header. The frame is discarded. Data 1: **Actual Length** Data1 Type: ushort Data 2: **Indicated Length** Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4: Code Location Data4 Type: ushort
Serial Port Reset Error	DNP[MCU]	Normal	An error occurred in reset of a serial port during a communicat configuration change. Changes to serial port configuration ma taken affect. Data 1: **Port Number** Data1 Type: PortCode Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
P2P Receive buffer full	DNP[MCU]	Normal	
P2P Fragment received from transport	DNP[MCU]	All	Data 1: **Peer Address** Data1 Type: ushort Data 2: **Fragment Length** Data2 Type: ushort

			Data 3: **Port ID** Data3 Type: ushort Data 4: **Fragment ID** Data4 Type: ushort
Error Registering Special Function	DNP[MCU]	Normal	
P2P Notification received	DNP[MCU]	All	Data 1: **Source Address / ID** Data1 Type: ushort Data 2: **Destination Address /Code** Data2 Type: ushort Data 3: **Sequence Number / Layer** Data3 Type: ushort Data 4: **Connection ID/0** Data4 Type: ushort
TCP Port Maintenance	DNP[MCU]	Extended	The actual length of the received DNP frame does not match tl indicated in the DNP data link header. The frame is discarded. Data 1: **TCP Port Connect State** Data1 Type: TCPPortConne Data 2: **TCP Port Active State** Data2 Type: TCPPortActiveS Data 3: **Destination RTU Address** Data3 Type: RTUAddress Data 4: **Diagnostic Value** Data4 Type: ushort
Invalid Transport Segment Length	DNP[MCU]	Extended	The transport segment length of the DNP frame is invalid. The discarded. The valid range is from 3 to 250. Data 1: **Segment Length** Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Receive Message Buffer Full	DNP[MCU]	Extended	The transport function receive message buffers are full. These the frame header information and other message details. The n discarded. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Receive Data Buffer Full	DNP[MCU]	Extended	The transport function receive data buffers are full. These buff data portion of the received DNP frames. The new frame is di Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Removed Old FIR-Only Message	DNP[MCU]	All	An incomplete fragment was discarded due to a newer messag received from the same source device. Data 1: **Transport Header** Data1 Type: DNPTransportHeade Data 2:Debug Data Data2 Type: ushort Data 3: **Source Address** Data3 Type: RTUAddress Data 4:Debug Data Data4 Type: ushort

App Layer Accepted FIR/FIN Message	DNP[MCU]	All	A single-frame fragment was successfully processed by the ap layer and is being removed from the transport function buffers Data 1: **Transport Header** Data1 Type: DNPTransportHeade Data 2: **Fragment Length** Data2 Type: ushort Data 3: **Source Address** Data3 Type: RTUAddress Data 4:Debug Data Data4 Type: ushort
App Layer Accepted FIN Only Message	DNP[MCU]	All	A multi-frame fragment was successfully processed by the app and is being removed from the transport function buffers. Data 1: **Transport Header** Data1 Type: DNPTransportHeade Data 2: **Fragment Length** Data2 Type: ushort Data 3: **Source Address** Data3 Type: RTUAddress Data 4:Debug Data Data4 Type: ushort
Frame Addition Unsuccessful	DNP[MCU]	Extended	A newly received frame of a multi-frame fragment was unable to the fragment due to the 2k fragment size restriction. The fra discarded. Data 1: **Transport Header of Fragment** Data1 Type: DNPTransportHeader Data 2: **Transport Header of Frame** Data2 Type: DNPTrans Data 3: **Source Address** Data3 Type: RTUAddress Data 4:Debug Data Data4 Type: ushort
Fragment Not Found	DNP[MCU]	Extended	A matching fragment was not found in the buffers for a newly frame of a multi-frame fragment. The frame was discarded. Data 1: **Transport Header** Data1 Type: DNPTransportHeade Data 2:Debug Data Data2 Type: ushort Data 3: **Source Address** Data3 Type: RTUAddress Data 4:Debug Data Data4 Type: ushort
Transp. Layer Transmit Fragment Too Long	DNP[MCU]	Extended	The application layer was requested to send a fragment larger size limitation. Data 1: **Fragment Length** Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Transp. Layer TX Fragment Buffer Full	DNP[MCU]	Extended	No free buffer was found in the transport function transmit buf application layer may save this message and reattempt to trans Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Frame Accepted by Data Link Layer	DNP[MCU]	All	A single frame was successfully handed off to the data link lay transmit. Note that this may be a single frame of a multi-frame Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Transport Header** Data3 Type: DNPTransportHeade Data 4: **Frame Length** Data4 Type: ushort
Message Timed Out on Receive List	DNP[MCU]	Extended	A fragment was removed from the transport function receive b extended period of inactivity. Data 1: **Transport Header** Data1 Type: DNPTransportHeade Data 2:Debug Data Data2 Type: ushort Data 3: **Source Address** Data3 Type: RTUAddress Data 4:Debug Data Data4 Type: ushort
Message Timed Out on Transmit List	DNP[MCU]	Extended	A fragment was removed from the transport function transmit an extended period of inactivity. Data 1: **Transport Header** Data1 Type: DNPTransportHeade Data 2:Debug Data Data2 Type: ushort Data 3: **Source Address** Data3 Type: RTUAddress Data 4:Debug Data Data4 Type: ushort
Removed Deferred Read Request	DNP[MCU]	Extended	A DNP read request that was deferred due to an outstanding u message was removed due to reception of a newer request. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Frame Declined by Datalink	DNP[MCU]	Extended	The datalink layer was unable to accept the request to transmit The frame transmit will be retried until it is accepted or the me out on the transmit list. Data 1: Error code returned by OS Data1 Type: ushort Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Transport Header** Data3 Type: DNPTransportHeade Data 4: **Frame Length** Data4 Type: ushort
Port Undefined in Transport	DNP[MCU]	All	Data 1: **Destination Addrss** Data1 Type: ushort
Reset Peer Link Sent	DNP[MCU]	Extended	A Reset data link frame was sent to a peer device in an attemp reinitialize peer-to-peer communications. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Reset Peer Link Received	DNP[MCU]	Extended	A Reset data link frame was received from the reported peer d Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Action Taken 1=record reinit 2=reset seq num only added** Data2 Type: DNPDLAction Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Message Taken Off Xmit List - No Peer	DNP[MCU]	Extended	The destination peer device was not found in the peer list, poss change in configuration. The message was discarded. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Application Control** Data3 Type: DNPAppControl Data 4: **Function Code** Data4 Type: DNPAppFunctionCode
Set URBE Timer Error	DNP[MCU]	Extended	An error was detected when attempting to start an internal time Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
No URBE Delay Timer	DNP[MCU]	Extended	No free timer was found when attempting to start an internal ti Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Duplicate Fragment Received	DNP[MCU]	Extended	The application layer detected reception of a duplicate fragmer previous response will be resent. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Application Control** Data3 Type: DNPAppControl Data 4:Debug Data Data4 Type: ushort
App.Layer.Transmi.tFragment.Too.Long	DNP[MCU]	Extended	The application layer was requested to send a fragment larger t size limitation. Data 1: **Fragment Length** Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
App. Layer TX Fragment Buffer Full	DNP[MCU]	Extended	No free buffer was found in the application layer transmit list. may be retried or discarded.

			Data 1: Code Location Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Invalid Fragment Length Received	DNP[MCU]	Extended	The application layer detected a fragment with an invalid lengt rand is from 2 to 2048 bytes. The fragment was discarded. Data 1: **Fragment Length** Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Incorrect Amount of Data Received	DNP[MCU]	Extended	The application layer detected an inconsistency in the amount fragment. The fragment was discarded. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Reported Length** Data2 Type: ushort Data 3: **Actual Length** Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Fragment Timed Out on Transmit List	DNP[MCU]	Extended	The fragment was removed from the application layer transmit extended period of inactivity. This will normally only occur if function and data link layer are unable to service transmit requ Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Application Control** Data3 Type: DNPAppControl Data 4: **Function Code** Data4 Type: DNPAppFunctionCode
App Layer Accepted Good Fragment	DNP[MCU]	All	The application layer successfully accepted a complete fragme transport function. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Data Length** Data2 Type: ushort Data 3: **Application Control** Data3 Type: DNPAppControl Data 4: **Function Code** Data4 Type: DNPTranspFunctionCod
Peer Device Buffer Full	DNP[MCU]	Extended	A peer device was unable to be added to the peer device list du buffer being full. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Peer Device Already on List	DNP[MCU]	Extended	Addition of the peer device to the list was unsuccessful becaus is already on the list. Data 1: **Peer RTU Address** Data1 Type: RTUAddress Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Peer Device Removed From List	DNP[MCU]	Extended	The peer device was successfully removed from the peer devic Data 1: **Peer RTU Address** Data1 Type: RTUAddress Data 2: Code Location Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Peer Device Added to List	DNP[MCU]	Extended	The peer device was successfully added to the peer device list. Data 1: **Peer RTU Address** Data1 Type: RTUAddress Data 2: **Connection ID** Data2 Type: ConnectionID Data 3: **Port Code** Data3 Type: PortCode Data 4:Master Number Data4 Type: Master
FIR/FIN Not Set	DNP[MCU]	Extended	The application layer found that the FIR and FIN bits of the A Control byte were not set. The fragment was discarded. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Application Control** Data3 Type: DNPAppControl Data 4:Debug Data Data4 Type: ushort
Sequence Number Mismatch	DNP[MCU]	Extended	The application layer found that the sequence number in the A Control byte was inconsistent with what was expected. The fra discarded. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Application Control** Data3 Type: DNPAppControl Data 4:Debug Data Data4 Type: ushort
Transient Peer Device Addition Issue	DNP[MCU]	Extended	The source of this fragment was unknown so an attempt was n to the peer device list. The attempt was unsuccessful so the fra discarded. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Application Control** Data2 Type: DNPAppControl Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
No Peer Record Found During Xmit	DNP[MCU]	Extended	A transmit was attempted to a device that was not found on the list. The transmit was aborted. Data 1: Code Location Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

URBE Registration Buffer Full	DNP[MCU]	Extended	The buffer containing callback functions for unsolicited messa when attempting to add another function. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Peer Device Not on List	DNP[MCU]	Extended	The requested peer device was not found on the peer device lis attempting to remove the device from the list. Data 1: **Peer RTU Address** Data1 Type: RTUAddress Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Initial Unsolicited Message Confirmed	DNP[MCU]	Extended	Indicates that a confirmation message was received for the uns messages that must be sent on power up. Data 1: **Callback Status** Data1 Type: CallbackStatus Data 2: **RTU Address** Data2 Type: RTUAddress Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
URBE Enabled Via SCADA	DNP[MCU]	Extended	Unsolicited Report by Exception processing was enabled over Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **URBE Status** Data2 Type: URBEStatus Data 3:Active URBE Class Mask Data3 Type: ActiveURBEC Data 4:Debug Data Data4 Type: ushort
URBE Disabled Via SCADA	DNP[MCU]	Extended	Unsolicited Report by Exception processing was disabled over Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **URBE Status** Data2 Type: URBEStatus Data 3:Active URBE Class Mask Data3 Type: ActiveURBEC Data 4:Debug Data Data4 Type: ushort
Port Code Invalid	DNP[MCU]	Extended	An invalid port code was detected when attempting to add a pe the peer device list. Data 1: **Peer RTU Address** Data1 Type: RTUAddress Data 2: **Invalid Port Code value** Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Cold Restart Requested	EVT[MCU]	Normal	A cold restart of the control has been requested over SCADA, performed in two seconds. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Function Code Not Implemented	DNP[MCU]	Extended	The application layer received a message containing a DNP fu that is invalid or is not implemented. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Function Code** Data3 Type: DNPAppFunctionCode Data 4:Debug Data Data4 Type: ushort
Unknown Master Access Restricted	DNP[MCU]	Extended	A device that was previously unknown has requested an actior restricted to configured master stations. The restricted actions select/operate and cold restart. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Function Code** Data3 Type: DNPAppFunctionCode Data 4:Debug Data Data4 Type: ushort
Peer Device Modified on List	DNP[MCU]	Extended	The peer device was successfully modified on the peer device Data 1: **Peer RTU Address** Data1 Type: RTUAddress Data 2: Code Location Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Master Number Data4 Type: ushort
Request time synchronization received	DNP[MCU]	All	A response was received with the need-time IIN set and a time being attempted. Data 1: **Status** Data1 Type: ushort Data 2: **Source Address** Data2 Type: RTUAddress Data 3: **Destination Address** Data3 Type: RTUAddress Data 4:Debug Data Data4 Type: ushort
P2P Bad transmit fragment length	DNP[MCU]	All	Data 1: **Length of Fragment** Data1 Type: ushort Data 2:Debug text Data2 Type: ushort Data 3:Debug text Data3 Type: ushort Data 4:Debug text Data4 Type: ushort
P2P Transmit fragment buffer full	DNP[MCU]	Normal	
P2P No peer device record	DNP[MCU]	All	Data 1: **Destination Address** Data1 Type: ushort
P2P Fragment declined by transport	DNP[MCU]	Extended	Data 1: **Result Code** Data1 Type: ushort Data 2: **Destination Address/Code** Data2 Type: ushort

			Data 3: **Fragment Length** Data3 Type: ushort Data 4: **Code Location** Data4 Type: ushort
P2P No peer for received fragment	DNP[MCU]	All	Data 1: **Peer Address** Data1 Type: ushort
P2P Application layer CRC error	DNP[MCU]	All	Data 1: **Source Address** Data1 Type: ushort Data 2: **Local Address** Data2 Type: ushort Data 3: **Expected Sequence Number** Data3 Type: ushort
P2P App layer sequence error	DNP[MCU]	Extended	Data 1: **Source Address** Data1 Type: ushort Data 2: **Connection ID** Data2 Type: ushort Data 3: **Expected sequence number** Data3 Type: ushort Data 4: **Received sequence number** Data4 Type: ushort
P2P Unknown peer access restricted	DNP[MCU]	Extended	Data 1: **Local Peer Addr** Data1 Type: ushort Data 2: **Source Peer Addr** Data2 Type: ushort Data 3: **Requested Function** Data3 Type: ushort Data 4: **Connection ID** Data4 Type: ushort
P2P Error block received from peer	DNP[MCU]	Normal	Data 1: **Source Addr** Data1 Type: ushort Data 2: **Connection ID** Data2 Type: ushort Data 3: **Errant Object** Data3 Type: ushort Data 4: **Errant Code** Data4 Type: ushort
P2P Function code not implemented	DNP[MCU]	All	Data 1: **Local Peer Addr** Data1 Type: ushort Data 2: **Source Peer Addr** Data2 Type: ushort Data 3: **Requested Function** Data3 Type: ushort Data 4: **Connection ID** Data4 Type: ushort
P2P Peer not P2P compatible	DNP[MCU]	All	Data 1: **Destination Address** Data1 Type: ushort Data 2: **Connection ID** Data2 Type: ushort
P2P Protocol not recognized	DNP[MCU]	All	Data 1: **Protocol Type** Data1 Type: ushort
P2P Error found in response data	DNP[MCU]	Extended	Data 1: **Source Address** Data1 Type: ushort

			Data 2: **Connection ID** Data2 Type: ushort Data 3: **Errant Object** Data3 Type: ushort Data 4: **Errant Code** Data4 Type: ushort
Association reset link sent	DNP[MCU]	All	Data 1: **Source Addr** Data1 Type: ushort Data 2: **Destination Addr** Data2 Type: ushort Data 3: **Port ID** Data3 Type: ushort
Association reset link send failed	DNP[MCU]	Extended	Data 1: **Source Addr** Data1 Type: ushort Data 2: **Destination Addr** Data2 Type: ushort Data 3: **Port ID** Data3 Type: ushort
Association reset link ack rcvd	DNP[MCU]	All	Data 1: **Peer Addr** Data1 Type: ushort Data 2: **Connection ID** Data2 Type: ushort
P2P Association test sent	DNP[MCU]	All	Data 1: **Destination Addr** Data1 Type: ushort
P2P Association test unsuccessful	DNP[MCU]	Extended	Data 1: **Destination Addr** Data1 Type: ushort
P2P Association test response rcvd	DNP[MCU]	All	Data 1: **Peer Device Address** Data1 Type: ushort Data 2: **P2P Support Flag** Data2 Type: ushort Data 3: **MTU Size** Data3 Type: ushort
P2P Support query received	DNP[MCU]	All	Data 1: **Source Address** Data1 Type: ushort
P2P Reset Association frame sent	DNP[MCU]	All	Data 1: **Peer Device Address** Data1 Type: ushort
Association negotiation incomplete	DNP[MCU]	Extended	Data 1: **Destination Address** Data1 Type: ushort Data 2: **Association State** Data2 Type: ushort
P2P Xmt temporarily suspended	DNP[MCU]	All	Data 1: **Destination Addr** Data1 Type: ushort Data 2: **Connection ID** Data2 Type: ushort Data 3: **Suspend Countdown Timer Value** Data3 Type: usho

P2P App layer fragment xmit	DNP[MCU]	All	Data 1: **Transmit Result Code** Data1 Type: ushort Data 2: **Destination Addr** Data2 Type: ushort Data 3: **Port ID** Data3 Type: ushort Data 4: **Fragment Length** Data4 Type: ushort
Local/Remote Callback Func. Registered	DNP[MCU]	Normal	
Special Function Buffer Full	DNP[MCU]	Extended	An attempt by the application to register a special memory rea function was rejected because the buffer was full. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Special Function Already on List	DNP[MCU]	Extended	An attempt by the application to register a special memory rea function was rejected because a special function for that memo already exists in the buffer. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error in Object Parse	DNP[MCU]	Extended	The header or data portion of an object that was returned in a r message was invalid or otherwise unexpected. If possible othe within the response will still be processed. Data 1: **Object Type** Data1 Type: DNPObjectType Data 2: **Source Address** Data2 Type: RTUAddress Data 3: **Destination Address** Data3 Type: RTUAddress Data 4: Data4 Type: ushort
Object Variation Error	DNP[MCU]	Extended	The variation of the object that was returned in the response m invalid or unsupported. Data 1: **Object Type** Data1 Type: DNPObjectType Data 2: **Source Address** Data2 Type: RTUAddress Data 3: **Destination Address** Data3 Type: RTUAddress Data 4: **Variation** Data4 Type: DNPObjectVariation
Object Qualifier Error	DNP[MCU]	Extended	The qualifier code of the object that was returned in the respor invalid or unsupported. Data 1: **Object Type** Data1 Type: DNPObjectType Data 2: **Source Address** Data2 Type: RTUAddress Data 3: **Destination Address** Data3 Type: RTUAddress Data 4: **Invalid Qualifier/ Index code** Data4 Type: ushort

Parsed Data Buffer Full	DNP[MCU]	Extended	The data buffer containing the parsed object data is full. No fu of this response message will take place. Data 1: **Object Type** Data1 Type: DNPObjectType Data 2: **Source Address** Data2 Type: RTUAddress Data 3: **Destination Address** Data3 Type: RTUAddress Data 4: **Variation** Data4 Type: DNPObjectVariation
Object Header Error	DNP[MCU]	Extended	An error was detected in the object header of the DNP messag Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Object Type** Data3 Type: DNPObjectType Data 4: **Variation** Data4 Type: DNPObjectVariation
Error Assembling Object Header	DNP[MCU]	Extended	An error was detected in the object header during assembly of message. The response process was aborted. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Destination Address** Data2 Type: RTUAddress Data 3: **Object Type** Data3 Type: DNPObjectType Data 4: **Variation** Data4 Type: DNPObjectVariation
Invalid Object Range Index	DNP[MCU]	Extended	An error was detected in an object header of the current reques related to the data index values. An attempt will be made to pr objects within the message. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **First Index or Index Size** Data2 Type: ushort Data 3: **Last Index or Qualifier** Data3 Type: ushort Data 4: Code Location Data4 Type: ushort
Fragment Data Size Error	DNP[MCU]	Extended	The amount of data available in the fragment was not consister expected amount of data. The fragment was discarded. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Data Length** Data2 Type: ushort Data 3: **Data Index** Data3 Type: ushort Data 4: Code Location Data4 Type: ushort
Object/Variation Not Supported	DNP[MCU]	Extended	The object being processed does not include a support object a combination. An attempt will be made to continue parsing oth within the message. Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Object Type** Data2 Type: DNPObjectType Data 3: **Variation** Data3 Type: DNPObjectVariation Data 4: Code Location Data4 Type: ushort
Special Function Registered	DNP[MCU]	Extended	A special function related to a specific memory action and virt address was successfully added to the list.

			Data 1: **Action** Data1 Type: VMFunctionCode Data 2: **Memory Address** Data2 Type: VAddr16 Data 3:Debug Data Data3 Type: VAddr16 Data 4:Debug Data Data4 Type: ushort
Virtual Memory Read or Write Error	DNP[MCU]	Extended	A request to read or write virtual memory addresses was unsuc Data 1: **Source Address** Data1 Type: RTUAddress Data 2: **Memory Address** Data2 Type: VAddr16 Data 3: Result Code returned by OS Data3 Type: ushort Data 4: **Action** Data4 Type: VMAccessAction
SBO Select Timer Unavailable	DNP[MCU]	Extended	An internal timer was not available to perform the select-befor timing function. The operation request was aborted. Data 1: **Object Type** Data1 Type: DNPObjectType Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
SBO Select Timer Error	DNP[MCU]	Extended	An error was detected when attempting to initialize a select-be timer. The operation request was aborted. Data 1: **Object Type** Data1 Type: DNPObjectType Data 2: **SBO Timer Value** Data2 Type: SBOTimerValue Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Issue With Output Block	DNP[MCU]	All	An operation request was unsuccessful due to an issue in the C Output Block of the message. See DNP documentation for a c of DNP Output Block status codes. Data 1: **Object Type** Data1 Type: DNPObjectType Data 2: **DNP Output Block Status Code** Data2 Type: DNPOutputBlockStatusCode Data 3: **Point Number** Data3 Type: ushort Data 4: **Function Code** Data4 Type: DNPAppFunctionCode
Control Point Function Registered	DNP[MCU]	Extended	A control point special function was successfully added to the function connects a control operation with functionality elsewl application. Data 1: **Point Code** Data1 Type: DNPPointCode Data 2: **Associated RTU Address** Data2 Type: RTUAddress Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Control Point Code Not Mapped	DNP[MCU]	Extended	A control point special function was not added to the list becau control point is not yet included in the point mapping configur

			Data 1: **Point Code** Data1 Type: DNPPointCode Data 2: **Associated RTU Address** Data2 Type: RTUAddress Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Change to Point Map Detected	DNP[MCU]	All	A configuration change to a point map was detected. This char processed 30 seconds after the last change is detected. Data 1: **Map Number** Data1 Type: MapNumber Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error in Getting Map Timer	DNP[MCU]	Extended	An error was detected when attempting to initialize a map-cha The operation will be retried. Data 1: **Map Number** Data1 Type: MapNumber Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Point Processing Timer Unavailable	DNP[MCU]	Extended	An internal timer was not available to perform the routing tabl configuration change timing function. DNP initialization will Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Point Map Reinitialized	DNP[MCU]	Normal	The point map was successfully reinitialized following comple changes to the point map configuration. Data 1: **Map Number** Data1 Type: MapNumber Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Point Definition Invalid	DNP[MCU]	Extended	An issue was found in the configuration data of a mapped poir will not be initialized until the error is corrected. Data 1: **Map Number** Data1 Type: MapNumber Data 2: **Point Index** Data2 Type: ushort Data 3: **Error Data** Data3 Type: ushort Data 4: Code Location Data4 Type: ushort
Input Point Code Not Mapped	DNP[MCU]	Extended	The application attempted to supply input data for a point that included in the configured point mapping. Data 1: **Point Type** Data1 Type: DNPPointType Data 2: **Point Code** Data2 Type: DNPPointCode

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Event Buffer Overflow	DNP[MCU]	Extended	An overflow condition has been reached for the event buffer o point type. The oldest event will be removed to make room for event, and the overflow IIN bit will be set. Data 1: **DNP Point Type** Data1 Type: DNPPointType Data 2:Master Number Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Analog Output Function Registered	DNP[MCU]	Extended	An analog output point special function was successfully adde list. This function connects an Analog output operation with f elsewhere in the application. Data 1: **Point Code** Data1 Type: DNPPointCode Data 2: **Associated RTU Address** Data2 Type: RTUAddress Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Analog Output Code Not Mapped	DNP[MCU]	Extended	An analog output point special function was not added to the li the analog output point is not yet included in the point mappin configuration. Data 1: **Point Code** Data1 Type: DNPPointCode Data 2: **Associated RTU Address** Data2 Type: RTUAddress Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Map Change Callback Buffer Full	DNP[MCU]	Extended	A special function to be used to inform the application about n was unable to be added to the list. The list was full. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Map Change Callback Registered	DNP[MCU]	Extended	A special function to be used to inform the application about n was successfully added to the list. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Route Processing Timer Unavailable	DNP[MCU]	Extended	An internal timer was not available to perform the map-change function. DNP initialization will be aborted. Data 1: **Map Number** Data1 Type: MapNumber Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Route Entry Invalid	DNP[MCU]	Extended	An issue was found with one of the configured route entries. T will not be initialized until the issue is corrected. Data 1: **Route RTU Address** Data1 Type: RTUAddress Data 2: **Route Table Index** Data2 Type: ushort Data 3: **Route Entry Error** Data3 Type: RouteEntryError Data 4:Debug Data Data4 Type: ushort
Route Could Not be Added	DNP[MCU]	Extended	The routing table did not have room for the new route. The rou not be initialized. Data 1: **Route RTU Address** Data1 Type: RTUAddress Data 2: **Route Table Index** Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Route Table Initialized	DNP[MCU]	Normal	The route table was successfully initialized with configured ro routing was enabled. Data 1: **Count of Routes** Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Route Configuration Change Detected	DNP[MCU]	All	A configuration change to the routing table was detected. This be processed 30 seconds after the last change is detected Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Error in Route Change Timer	DNP[MCU]	Extended	An error was detected when attempting to initialize a route tab timer. The operation will be retried. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Pass-Through Routing Enabled	DNP[MCU]	Normal	A valid pass-through route configuration was found and pass-t routing was enabled. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Pass-Through Routing Disabled	DNP[MCU]	Normal	A valid pass-through route configuration was not found and pa routing was disabled. Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Diag Test Start	DNP[MCU]	Normal	Data 1: **Test Type** Data1 Type: DiagTestType Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Diag KeepAlive Start	DNP[MCU]	Normal	Data 1: **Test Type** Data1 Type: DiagTestType Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Diag Test Ending	DNP[MCU]	Normal	Data 1: **Test Type** Data1 Type: DiagTestType Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Diag KeepAlive End	DNP[MCU]	Normal	Data 1: **Test Type** Data1 Type: DiagTestType Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Diag Ping Timeout	DNP[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNPDiag Send Error	DNP[MCU]	Normal	Data 1: **DNP Address** Data1 Type: DNPAddress Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Diag Max Peers	DNP[MCU]	Normal	

			Data 1: **Max Peers When This Occurs** Data1 Type: MaxPeer Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Diag No Peers	DNP[MCU]	Normal	Data 1: **No Peers When This Occurs** Data1 Type: NoPeersW Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNPDiag Test Suspended	DNP[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNPDiag Test Resumed	DNP[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNPDiag WatchDog Triggered	DNP[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Comm Sys Alarm On	DNP[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
DNP Comm Sys Alarm Off	DNP[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
No Diagnostics On Peer. Cannot Test.	DNP[MCU]	All	Remote Peer does not have diagnostics installed, so tests cann performed with it. Data 1: **DNP Address** Data1 Type: DNPAddress Data 2:Debug Data Data2 Type: ushort

			Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switching to Failover Master	DNP[MCU]	Extended	Switching to Failover Master Data 1: **Master RTU Address** Data1 Type: RTUAddress Data 2: **IP addr first half** Data2 Type: Hexushort Data 3:IP addr second half Data3 Type: Hexushort Data 4:Master Number Data4 Type: ushort
Switching to Master Primary	DNP[MCU]	Extended	Switching to Master Primary Data 1: **Master RTU Address** Data1 Type: RTUAddress Data 2: **IP addr first half** Data2 Type: Hexushort Data 3:IP addr second half Data3 Type: Hexushort Data 4:Master Number Data4 Type: ushort
Master Wrong IP Addr Cmd Rejected	DNP[MCU]	Extended	Switching to Master Primary Data 1: **Master RTU Address** Data1 Type: RTUAddress Data 2: **IP addr first half** Data2 Type: Hexushort Data 3:IP addr second half Data3 Type: Hexushort Data 4:Port Code Data4 Type: ushort
Invalid Master Cmd Rejected	DNP[MCU]	Extended	Switching to Master Primary Data 1: **Master RTU Address** Data1 Type: RTUAddress Data 2: **IP addr first half** Data2 Type: Hexushort Data 3:IP addr second half Data3 Type: Hexushort Data 4:Function Code Data4 Type: Hexushort
Master has wrong port type Cmd Rejected	DNP[MCU]	Extended	Switching to Master Primary Data 1: **Master RTU Address** Data1 Type: RTUAddress Data 2: **IP addr first half** Data2 Type: Hexushort Data 3:IP addr second half Data3 Type: Hexushort Data 4:Port Code Data4 Type: ushort
Master used reserved addr. Cmd Rejected	DNP[MCU]	Extended	Switching to Master Primary Data 1: **Master RTU Address** Data1 Type: RTUAddress Data 2: **IP addr first half** Data2 Type: Hexushort Data 3:IP addr second half Data3 Type: Hexushort Data 4:Port Code Data4 Type: ushort
SCADA Set Date/Time	DNP[MCU]	All	SCADA Set Date/Time Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Recorded Time is 0	DNP[MCU]	All	Recorded Time is 0 Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Cold Restart Command Disabled	DNP[MCU]	All	Cold Restart Command Disabled Data 1:Debug Data Data1 Type: RTUAddress Data 2:Debug Data Data2 Type: RTUAddress Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Disk Error	CFM[MCU]	All	Data 1:Trace Info Data1 Type: ushort Data 2:Error Code Data2 Type: ushort
File Allocation In Progress	CFM[MCU]	Normal	
File Allocation Not In Progress	CFM[MCU]	Normal	
File Allocation Issue	CFM[MCU]	All	Data 1:Trace Info Data1 Type: ushort Data 2:Error Code Data2 Type: ushort
Compact Flash Card Not Found	CFM[MCU]	Extended	Data 1:Debug Info (Card State) Data1 Type: ushort
Disk Check Issue	CFM[MCU]	All	Data 1:Error Code Data1 Type: ushort
Set Bad Disk Condition to On	CFM[MCU]	Normal	
Set Bad Disk Condition to Off	CFM[MCU]	Normal	
Allocated File Size Exceeded	CFM[MCU]	All	Data 1:File Class Data1 Type: ushort Data 2: **Allocated Size (kB)** Data2 Type: ushort Data 3: **Written Size (kB)** Data3 Type: ushort
Shutdown Processing CFM	CFM[MCU]	Normal	Data 1:Trace Info Data1 Type: ushort
CFM:Disk Tampered Indication	CFM[MCU]	Normal	Directory Contents Corruption Was Detected on Startup.
CFM:Disk Tampered Indication Cleared	CFM[MCU]	Normal	Directory Contents Corruption Was Detected on Startup.
SW: Control Open Request	IIM[MCU]	Normal	
SW: Control Open OK	IIM[MCU]	Normal	
SW: Control Close Request	IIM[MCU]	Normal	

SW: Control Close OK	IIM[MCU]	Normal	
SW2: Control Open Request	IIM[MCU]	Normal	
SW2: Control Open OK	IIM[MCU]	Normal	
SW2: Control Close Request	IIM[MCU]	Normal	
SW2: Control Close OK	IIM[MCU]	Normal	
Control Operation Request Problem	IIM[MCU]	Normal	Data 1: Data1 Type: ControlCode
Control Operation Problem	IIM[MCU]	Normal	Data 1: Data1 Type: ControlCode
Average Load Reset	IIM[MCU]	Normal	Data 1: Data1 Type: AverageLoad Data 4: Data4 Type: ushort
SCADA Cleared Manual Operation	IIM[MCU]	Normal	
SW: Any Phase Overcurrent	IIM[MCU]	Normal	Data 1: Data1 Type: FaultBits
SW: Overcurrents Cleared	IIM[MCU]	Normal	
SW: Any Phase Loss Active	IIM[MCU]	Normal	Data 1: Data1 Type: AnyPhaseLoss
SW: Any Phase Loss Cleared	IIM[MCU]	Normal	
SW1: Manual Operation Detected	IIM[MCU]	Normal	
SW: Manual Operation Cleared	IIM[MCU]	Normal	
SW: Hot-Line Tag Active	IIM[MCU]	Normal	
SW: Hot-Line Tag Cleared	IIM[MCU]	Normal	
Frequency Trip Active	IIM[MCU]	Normal	Data 1: Data1 Type: FreqTripType
Frequency Trip Cleared	IIM[MCU]	Normal	
SW: Trip to Lockout Active	IIM[MCU]	Normal	
SW: Trip to Lockout Cleared	IIM[MCU]	Normal	
SW1 Opened	IIM[MCU]	Normal	
SW1 Closed	IIM[MCU]	Normal	
SW2 Open	IIM[MCU]	Normal	
SW2 Closed	IIM[MCU]	Normal	
Device In Reset State	IIM[MCU]	Normal	
Device Not In Reset State	IIM[MCU]	Normal	

Inst. Replay Status Update	IIM[MCU]	All	Data 1:Sw. Status Data1 Type: ushort Data 2:OC Status Data2 Type: ushort Data 3:VL Status Data3 Type: ushort
Manual Operation Report	IIM[MCU]	Normal	
SW2: Any Phase Overcurrent	IIM[MCU]	Normal	Data 1: Data1 Type: FaultBits
SW2: Overcurrents Cleared	IIM[MCU]	Normal	
SW2: Any Phase Loss Active	IIM[MCU]	Normal	
SW2: Any Phase Loss Cleared	IIM[MCU]	Normal	
SW2: Manual Operation Detected	IIM[MCU]	Normal	
SW2: Manual Operation Cleared	IIM[MCU]	Normal	
SW2: Hot Line Tag Activated	IIM[MCU]	Normal	
SW2: Hot Line Tag Cleared	IIM[MCU]	Normal	
SW2: Trip to Lockout Active	IIM[MCU]	Normal	
SW2: Trip to Lockout Cleared	IIM[MCU]	Normal	
System Voltage Unrecognized Active	IIM[MCU]	All	System Voltage Unrecognized Active Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
System Voltage Unrecognized Inactive	IIM[MCU]	All	System Voltage Unrecognized Inactive Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch 1 not in normal state inactive	DAT[MCU]	All	Switch 1 not in normal state inactive Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Switch 1 not in normal state active	DAT[MCU]	All	Switch 1 not in normal state active Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort

Local Switch is the Closed Loop Center	DAT[MCU]	All	Switch 1 not in normal state active Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
Local Switch no longer Closed Loop Ctr	DAT[MCU]	All	Switch 1 not in normal state inactive Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ushort
RTL Xmit Blocked	DAT[MCU]	All	Data 1:Debug Data Data1 Type: ManualOp Data 2:Debug Data Data2 Type: HLT Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: NoMaster
Xfer Trip PR Initiated (DG POI) Active	DAT[MCU]	Normal	Xfer Trip PR Initiated (DG POI) Active Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Xfer Trip PR Initiated (DG POI) Inactive	DAT[MCU]	Normal	Xfer Trip PR Initiated (DG POI) Inactive Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Transfer Trip Sent Active	DAT[MCU]	Normal	Transfer Trip Sent Active Data 1: Data1 Type: RTUAddress Data 2: Data2 Type: SWPos Data 3: Data3 Type: RTUAddress Data 4: Data4 Type: ushort
Transfer Trip Sent Inactive	DAT[MCU]	Normal	Transfer Trip Sent Inactive Data 1: Data1 Type: RTUAddress Data 2: Data2 Type: SWPos Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
DG Reconnect Disqualified On Fault	DAT[MCU]	Normal	DG Reconnect Disqualified On Fault Data 1: Data1 Type: TeamNumberWithUnknown

			Data 2: Data2 Type: BooleanTrueFalse Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Sw1 Auto Manl Op. Clear Timer Stopped	DAT[MCU]	Normal	Sw1 Auto Manual Operation Clearing Timer Stopped Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Sw2 Auto Manl Op. Clear Timer Stopped	DAT[MCU]	Normal	Sw2 Auto Manual Operation Clearing Timer Stopped Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Sw2 Auto Manl Op. Clear Timer Started	DAT[MCU]	Normal	Sw2 Auto Manual Operation Clearing Timer Started Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Auto Operation Enabled	FPX[MCU]	Normal	Auto Operation Enabled
Auto Operation Disabled	SWX[MCU]	Normal	Auto Operation Disabled
Cabinet Door Closed	FPX[MCU]	Normal	Cabinet Door Closed
Cabinet Door Open	SWX[MCU]	Normal	Cabinet Door Open
SCADA Mode - Remote	FPX[MCU]	Normal	SCADA Mode - Remote
SCADA Mode - Local	SWX[MCU]	Normal	SCADA Mode - Local
Auto Restoration Enabled	FPX[MCU]	Normal	Auto Restoration ON
Auto Restoration Disabled	SWX[MCU]	Normal	Auto Restoration OFF
USER1 Mode Enabled	FPX[MCU]	Normal	USER1 Mode ON
USER1 Mode Blocked	SWX[MCU]	Normal	USER1 Mode OFF
USER2 Mode Enabled	FPX[MCU]	Normal	USER2 Mode ON
USER2 Mode Blocked	SWX[MCU]	Normal	USER2 Mode OFF
SW Auto Operation Enabled	FPX[MCU]	Normal	SW Auto Operation Enabled
SW Auto Operation Disabled	SWX[MCU]	Normal	SW Auto Operation Disabled
SW2 Auto Operation Enabled	FPX[MCU]	Normal	SW2 Auto Operation Enabled
SW2 Auto Operation Disabled	SWX[MCU]	Normal	SW2 Auto Operation Disabled
Switch Trouble	FPX[MCU]	Normal	Switch Trouble
Switch OK	SWX[MCU]	Normal	Switch Trouble Vanishes
Use Alternate Reclose Count	FPX[MCU]	Normal	Use Alternate Reclose Count
Use Regular Reclose Count	SWX[MCU]	Normal	Use Regular Reclose Count

Issued Clr Electronics Bad Cmd	FPX[MCU]	Normal	Issued Clr Electronics Bad Cmd
Align Mode On	FPX[MCU]	Normal	Align Mode On
Align Mode Off	FPX[MCU]	Normal	Issued Clr Electronics Bad Cmd
Set Limits Mode On	FPX[MCU]	Normal	Set Limits Mode On
Set Limits Mode Off	FPX[MCU]	Normal	Set Limits Mode Off
SW Phase A Overcurrent On	FLT[MCU]	Normal	SW Phase A Overcurrent Detected
SW Phase A Overcurrent Off	FLT[MCU]	Normal	SW Phase A Overcurrent Cleared
SW Phase B Overcurrent On	FLT[MCU]	Normal	SW Phase B Overcurrent Detected
SW Phase B Overcurrent Off	FLT[MCU]	Normal	SW Phase B Overcurrent Cleared
SW Phase C Overcurrent On	FLT[MCU]	Normal	SW Phase C Overcurrent Detected
SW Phase C Overcurrent Off	FLT[MCU]	Normal	SW Phase C Overcurrent Cleared
SW Neutral Overcurrent ON	FLT[MCU]	Normal	SW Neutral Overcurrent Detected
SW Neutral Overcurrent OFF	FLT[MCU]	Normal	SW Phase A Overcurrent Cleared
SW2 Phase A Overcurrent ON	FLT[MCU]	Normal	SW2 Phase A Overcurrent Detected
SW2 Phase A Overcurrent OFF	FLT[MCU]	Normal	SW2 Phase A Overcurrent Cleared
SW2 Phase B Overcurrent ON	FLT[MCU]	Normal	SW2 Phase B Overcurrent Detected
SW2 Phase B Overcurrent OFF	FLT[MCU]	Normal	SW2 Phase B Overcurrent Cleared
SW2 Phase C Overcurrent ON	FLT[MCU]	Normal	SW2 Phase C Overcurrent Detected
SW2 Phase C Overcurrent OFF	FLT[MCU]	Normal	SW2 Phase C Overcurrent Cleared
SW2 Neutral Overcurrent ON	FLT[MCU]	Normal	SW2 Neutral Overcurrent Detected
SW2 Neutral Overcurrent OFF	FLT[MCU]	Normal	SW2 Neutral Overcurrent Cleared
SW Overcurrent ON	FLT[MCU]	Normal	SW Overcurrent Detected
SW Overcurrent OFF	FLT[MCU]	Normal	SW Overcurrent Cleared
SW2 Overcurrent ON	FLT[MCU]	Normal	SW2 Overcurrent Detected
SW2 Overcurrent OFF	FLT[MCU]	Normal	SW2 Overcurrent Cleared
ABC Loss of Voltage On	FLT[MCU]	Normal	SW Loss of Voltage Detected
ABC Loss of Voltage Off	FLT[MCU]	Normal	SW Voltage Restored
DEF Loss of Voltage ON	FLT[MCU]	Normal	SW2 Loss of Voltage Detected
DEF Loss of Voltage OFF	FLT[MCU]	Normal	SW2 Voltage Restored
SW Sectionalizer Trip ON	FLT[MCU]	Normal	SW Sectionalizer Triped
SW Sectionalizer Trip OFF	FLT[MCU]	Normal	SW Sectionalizer Trip Cleared
SW2 Sectionalizer Trip ON	FLT[MCU]	Normal	SW2 Sectionalizer Triped
SW2 Sectionalizer Trip OFF	FLT[MCU]	Normal	SW2 Sectionalizer Trip Cleared
SW Reverse Current A ON	FLT[MCU]	Normal	SW Reverse Current Phase A Detected
SW Reverse Current A OFF	FLT[MCU]	Normal	SW Phase A Normal Current Direction
SW Reverse Current B ON	FLT[MCU]	Normal	SW Reverse Current Phase B Detected
SW Reverse Current B OFF	FLT[MCU]	Normal	SW Phase B Normal Current Direction
SW Reverse Current C ON	FLT[MCU]	Normal	SW Reverse Current Phase C Detected
SW Reverse Current C OFF	FLT[MCU]	Normal	SW Phase C Normal Current Direction

SW2 Reverse Current A ON	FLT[MCU]	Normal	SW Reverse Current Phase A Detected
SW2 Reverse Current A OFF	FLT[MCU]	Normal	SW2 Phase A Normal Current Direction
SW2 Reverse Current B ON	FLT[MCU]	Normal	SW Reverse Current Phase B Detected
SW2 Reverse Current B OFF	FLT[MCU]	Normal	SW Phase B Normal Current Direction
SW2 Reverse Current C ON	FLT[MCU]	Normal	SW2 Reverse Current Phase C Detected
SW2 Reverse Current C OFF	FLT[MCU]	Normal	SW2 Phase C Normal Current Direction
Volt Imbal Det On	FLT[MCU]	Normal	Voltage Imbalance Detect On
Volt Imbal Det Off	FLT[MCU]	Normal	Voltage Imbalance Detect Off
Voltage Lost on Phases	FLT[MCU]	Normal	One or More Phases Voltage Lost Data 1: **SW Voltage Lost on** Data1 Type: SW1LostPhases Data 2: **SW2 Voltage Lost on** Data2 Type: SW2LostPhases
Voltage OK on All Phases	FLT[MCU]	Normal	Voltage Returned on All Phases
OC then Volt OK: Load-side Protect Open	FLT[MCU]	Normal	OC then voltage o.k.; Load-side protective open;Noted
OC then VL; Source Side Open; Counting	FLT[MCU]	Normal	OC then VL; Source-side protective open;Counting
Reclose Sequence Ended (time limit)	FLT[MCU]	Normal	Reclose memory time limit; Sequence ended; Count reset
Volt Loss; Source Side Fault; Sect OFF	FLT[MCU]	Normal	VL; Source-side fault; Sectionalizing disarmed
Volt Loss; Load-side fault; Sect ON	FLT[MCU]	Normal	VL; Load-side fault; Sectionalizing armed
Full Count Reached; Source Side Fault	FLT[MCU]	Normal	Full count reached; Source-side fault; Noted
Full Count Reached; Load Side Fault	FLT[MCU]	Normal	Full count reached; Load-side fault; Noted
Full Count Reached w/ Sect. OFF	FLT[MCU]	Normal	Full count reached w/ sectional. disabled; Noted
Full count reached; Open executed	FLT[MCU]	Normal	Full count reached; Open operation executed
No OC before VL; S-Side Open; Counting	FLT[MCU]	Normal	No OC before VL; Source Side open; Counting
Source-side Fuse Blown Load-side Fault	FLT[MCU]	Normal	OC then VL; Source-side fuse blown for load-side fault
Source-side Fuse Blown Source-side Fault	FLT[MCU]	Normal	No OC then VL; Source-side (SS) fuse blown - SS fault
CLOSE Exec Shots-to-Lockout Requested	FLT[MCU]	Normal	CLOSE operation executed, shots-to-lockout requested
Lockout Close w/ Event - OPEN Exec	FLT[MCU]	Normal	Lockout close complete w/ event after - OPEN executed
Persistent Phase Imbalance - OPEN Exec	FLT[MCU]	Normal	Persistent phase imbalance; OPEN executed
Phase Imbalance w/ Reclose Enabled	FLT[MCU]	Normal	Phase imbalance w/ reclose enabled; Waiting
Switch Closed; Reclose Cancelled	FLT[MCU]	Normal	Switch closed; Operator action; Reclose cancelled
Voltage Restored after Imbalance	FLT[MCU]	Normal	Voltage OK after imbalance; Voltage restored; Waiting
Switch Closed; Reclose Cancelled	FLT[MCU]	Normal	Switch closed; Operator action; Reclose cancelled (duplicate?)
Imbalance Corrected RECLOSE Exec	FLT[MCU]	Normal	Imbalance corrected w/ reclose enabled; CLOSE executed
Shots-to-Lockout Latched On	FLT[MCU]	Normal	Shots-to-lockout latched on

Successful Reclose; Sequence Ended	FLT[MCU]	Normal	Successful reclose; Sequence ended; Count reset
LOV Sectionalizing Fail - Sync Error	FLT[MCU]	Normal	LOV Fail Sync Err
FI Breaker Tripped	FLT[MCU]	Normal	FI Breaker Tripped
FI Breaker Closed	FLT[MCU]	Normal	FI Breaker Closed
BP Volt Loss w/ Current Present	FLT[MCU]	Normal	No OC then VL; Current still present; Noted
Volt Loss Only Count Reached	FLT[MCU]	Normal	No OC before VL; Volt Loss Only count reached
OPEN Exec on Volt Loss Only	FLT[MCU]	Normal	Open operation executed on Voltage Loss Only
Sect Disabled for Voltage Loss Only	FLT[MCU]	Normal	Sectionalizing Disabled on Voltage Loss Only; None
Extended Volt Loss; OPEN	FLT[MCU]	Normal	Reclose memory time limit; Extended Volt Loss; OPEN
Instant reclose Qualification w/ OC	FLT[MCU]	Normal	O/C then VLoss; Qualify for instant reclose
Instant reclose Qualification No OC	FLT[MCU]	Normal	No O/C & VLoss; Qualify for instant reclose
Persistent VLoss; Reclose Qualified	FLT[MCU]	Normal	Persistent VLoss; Breaker Operation Qualified
Instant Reclose - Not Counted	FLT[MCU]	Normal	OC + Instant Reclose
Tripped on Phase Loss	FLT[MCU]	Normal	Tripped based on Phase Loss
SW Ph. Loss Prot. LOV Phase A	FLT[MCU]	Normal	SW Ph. Loss Prot. LOV Phase A
SW Ph. Loss Prot. LOV Phase B	FLT[MCU]	Normal	SW Ph. Loss Prot. LOV Phase B
SW Ph. Loss Prot. LOV Phase C	FLT[MCU]	Normal	SW Ph. Loss Prot. LOV Phase C
SW Ph. Loss Prot. LOV Phase D	FLT[MCU]	Normal	SW Ph. Loss Prot. LOV Phase D
SW Ph. Loss Prot. LOV Phase E	FLT[MCU]	Normal	SW Ph. Loss Prot. LOV Phase E
SW Ph. Loss Prot. LOV Phase F	FLT[MCU]	Normal	SW Ph. Loss Prot. LOV Phase F
Originating a SRC Runner	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Received Old Runner	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Received Duplicate Runner	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Received New Runner	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort

			Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Forwarding a Received Runner	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Originating an Endload Runner	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
COS Cleared	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Team Config Netlist Mismatch	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
COS Active	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Missing Runners	NET[MCU]	Normal	Data 1: **This state provides a user programmable time dela** **starting the runners repeated after each successful round o** **+ distribution** Data1 Type: StatusFeedbackArea Data 2: **Runner's Quantity** Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
All Runners Received	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort

			Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETLIST no alt source found	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETLIST alt source result	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Building COS Runner	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
COS Reported by IT	NET[MCU]	All	Data 1: **Local Load COS** Data1 Type: ushort Data 2: **Average Load** Data2 Type: ushort Data 3: **3-Phase Average Load** Data3 Type: ushort Data 4: **Team Member Number** Data4 Type: TeamRecord
COS Received in Runner	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Forwarding Dup. COS Runner	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Loc Netlist Old. Need it From RTU	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort

New Netlist Request From RTU	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Netlist Record Sent	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Netlist Record Received	NET[MCU]	All	Data 1:Instance Data1 Type: Instance Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
ITII Mode is Active	NET[MCU]	Normal	Data 1:Instance Data1 Type: Instance Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
ITII Mode is Inactive	NET[MCU]	Normal	Data 1:Instance Data1 Type: Instance Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NET: Missing Runners in Adj Net Active	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NET: Missing Runners Adj Net Inactive	NET[MCU]	All	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX Adj data delivery runner timeout	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort

			Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX Adj data delivery runner initiated	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX Indiv Adj Data Runner Path Timeout	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX Indiv Adj Data Runner Path Present	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETV: Netlist Changed	NET[MCU]	Normal	Data 1: **Netview number of devices/switches in DivisionNet** NumOfDevices Data 2: **Number of entries in the D-W (Device-Wire) tables** NumOfWirePairs Data 3: **Number of Teams** Data3 Type: NumOfTeams Data 4:Null Data4 Type: ushort
NETV: Network Config. Changed	NET[MCU]	Normal	Data 1: **Number of rows in table for DivisionNet** Data1 Type NumOfRows Data 2: **Number of power sources in DivisionNet** Data2 Typ NumOfPowSrc Data 3: **Net View Analyzer** Data3 Type: NetViewAnalyzer Data 4:Number is there for debugging use Data4 Type: ushort
NETX dispatch intiated	NET[MCU]	Normal	Data 1: **Net Object Type** Data1 Type: RunnerType Data 2: **Source RTU Address** Data2 Type: RTUAddress Data 3: **Source Node ID** Data3 Type: SrcNodeID Data 4: **Destination Node ID** Data4 Type: DestNodeID
NETX EPD netlist received	NET[MCU]	Normal	

			Data 1: **Net Object ID** Data1 Type: ObjID Data 2: **Size of the arriving object in bytes, also a flag/signa to State machine A** Data2 Type: SizeOfObject Data 3: **Net Object Length** Data3 Type: ObjectLength Data 4: **Bufsiz** Data4 Type: bufsiz
NETX Net list trouble	NET[MCU]	Normal	Data 1: **Net Object ID** Data1 Type: ObjID Data 2: **Size of the arriving object in bytes, also a flag/signa to State machine A** Data2 Type: SizeOfObject Data 3: **Net Object Length** Data3 Type: ObjectLength Data 4: **Bufsiz** Data4 Type: bufsiz
NETX bad runner index	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
NETX node index not found	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Node Index** Data3 Type: NodeIndex Data 4: **Net Object Type** Data4 Type: RunnerType
NETX bad object data length	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object Length** Data2 Type: ObjectLength Data 3: **Data Length** Data3 Type: DataLength Data 4: **Net Object Type** Data4 Type: RunnerType
NETX new netlist delivery	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4: **Size of runner in RTU addresses** Data4 Type: RTUA
NETX chk for comm chk runner's passing	NET[MCU]	Normal	NETX check for communication check runner's passing Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4: **Feeder Net ID** Data4 Type: FeederNetID
NETX chk for comm chk runner's return	NET[MCU]	Normal	NETX check for communication check runner's return Data 1: **Source RTU Address** Data1 Type: RTUAddress

			Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4: **Feeder Net ID** Data4 Type: FeederNetID
NETX intiate communication check runner	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NEX communication check runner's timeout	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX communication check runner complete	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX activation runner return	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4: **Size of runner in RTU addresses** Data4 Type: RTUA
NETX activation of the runner's passing	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4: **Size of runner in RTU addresses** Data4 Type: RTUA
NETX activation of runner's intiation	NET[MCU]	Normal	Data 1: **RETV:AT EPD Launch all the Activation Runners RTU in line** Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX activation of runner is complete	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort

			Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX activation of runner's timout	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX collection runner return	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4: **FeederNetID** Data4 Type: FeederNetID
NETX collection runner's passing	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4: **FeederNetID** Data4 Type: FeederNetID
NETX collection as the runner intiates	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX collection of runner's timeout	NET[MCU]	Normal	Data 1: **RETV:AT EPD Launch all the Activation Runners RTU in line** Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX data delivery runner's return	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4: **FeederNetID** Data4 Type: FeederNetID
NETX data delivery of runner's passing	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4: **FeederNetID** Data4 Type: FeederNetID

NETX data delievery runner is intitiated	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX data delivery of runner's timeout	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX Feeder Net delivery to intiate rnr	NET[MCU]	Normal	NETX Feeder Net delivery to intiate runner Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX Feeder Net runner delivery complete	NET[MCU]	Normal	>NETX Feeder Net delivery of runner is complete Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX Feeder Net delivery rnr's timeout	NET[MCU]	Normal	NETX Feeder Net delivery runner's timeout Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX data runner's cycle is complete	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX continue adjacent distributrers	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination Node ID** Data2 Type: DestNodeID Data 3: **Runner Number** Data3 Type: RunnerNumber Data 4:Null Data4 Type: ushort
NETX inti adj Feeder Net distributers	NET[MCU]	Normal	NETX intiate adjacent Feeder Net distributers Data 1: **Source RTU Address** Data1 Type: RTUAddress

			Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4: **Net Object Length** Data4 Type: ObjectLength
NETX init adjacent data block reporter	NET[MCU]	Normal	NETX intiate adjacent data block reporter Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4: **Net Object Length** Data4 Type: ObjectLength
NETX fwd adjacent data block reporter	NET[MCU]	Normal	NETX forward adjacent data block reporter Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4: **Net Object Length** Data4 Type: ObjectLength
NETX sync net list to DAT delayed	NET[MCU]	Normal	Data 1: **SyncStat** Data1 Type: syncstat Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX sync net list to DAT is not good	NET[MCU]	Normal	Data 1: **SyncStat** Data1 Type: syncstat Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX netlist copy request	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net object ID** Data2 Type: ObjID Data 3: **Feeder Net ID** Data3 Type: FeederNetID Data 4: **Feeder Net CRC** Data4 Type: FeederNetCRC
NETX netlist copy response	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Feeder Net ID** Data3 Type: FeederNetID Data 4: **Feeder Net CRC** Data4 Type: FeederNetCRC
NETX Feeder Net ID and CRC mismatch	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4: **Net Object Type** Data4 Type: RunnerType

NETX netlist reporter returned	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4: **Object Length** Data4 Type: ObjectLength
NETX forward net list reporter	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4: **Object Length** Data4 Type: ObjectLength
NETX net list copy intiate	NET[MCU]	Normal	Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Feeder Net CRC** Data2 Type: FeederNetCRC Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX report adj netlist distributers	NET[MCU]	Normal	NETX report adjacent netlist distributers Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX report adjacent data block	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Configuration is in progress	NET[MCU]	Normal	Data 1: **Instance** Data1 Type: Instance Data 2: **Source RTU Address** Data2 Type: RTUAddress Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4: **Feeder Net ID** Data4 Type: FeederNetID
Configuration is not in progress	NET[MCU]	Normal	Data 1: **Previous State** Data1 Type: StatusFeedbackArea Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
Settings Accepted	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort

			Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Settings Received	NET[MCU]	Normal	If Data 4 reads 10 then Data 4 = Net Object State Revived Cor Data 1: **Net View number of devices in DivisionNet** Data1 T Data 2: **Error Code 1** Data2 Type: Hexushort Data 3: **Error Code 2** Data3 Type: Hexushort Data 4:Null Data4 Type: ushort
Netlist Propagation Enabled	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Netlist Propagation Disabled	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Netlist Loaded From BMM	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Netlist BMM Load Issue	NET[MCU]	Normal	Data Field 1 could be the BMM_Result in which case please r "BMMRESULT" DataType Data 1: **Bmm_State** Data1 Type: BMMSTATE Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Netlist Saved To BMM	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Netlist BMM Save Issue	NET[MCU]	Normal	Data Field 1 could be the BMM_Result in which case please r "BMMRESULT" DataType Data 1: **Bmm_State** Data1 Type: BMMSTATE Data 2:Null Data2 Type: ushort

			Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Netlist Was Erased	NET[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort
NETO buffer pool intialized	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETO combined buffers	NET[MCU]	Normal	Data 1: **Loop Count** Data1 Type: ushort Data 2: **Del Count** Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETO buffer pool length is bad	NET[MCU]	Normal	Data 1: **Total Length** Data1 Type: totallen Data 2: **Pool Size** Data2 Type: ushort Data 3: **Cumulative Free Buffer Space** Data3 Type: netobuff Data 4: **Loop Count** Data4 Type: ushort
NETO does not enough space to store obj	NET[MCU]	Normal	NETO does not have enough space to store object Data 1: **Net Fragment ID** Data1 Type: netfragID Data 2: **Source Net ID** Data2 Type: SrcNodeID Data 3: **Cumulative Free Buffer Space** Data3 Type: netobuff Data 4: **Data Length** Data4 Type: DataLength
NETO object buffering is full	NET[MCU]	Normal	Data 1: **Net Fragment ID** Data1 Type: netfragID Data 2: **Source Net ID** Data2 Type: SrcNodeID Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETO bad store request	NET[MCU]	Normal	Data 1: **Net Fragment ID** Data1 Type: netfragID Data 2: **Source Net ID** Data2 Type: SrcNodeID Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETO data buffer is full	NET[MCU]	Normal	Data 1: **Net Fragment ID** Data1 Type: netfragID Data 2: **Source Net ID** Data2 Type: SrcNodeID

			Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETO object handed up	NET[MCU]	Normal	Data 1: **Net Object State** Data1 Type: ObjState Data 2: **Source RTU Address** Data2 Type: RTUAddress Data 3: **Net Object ID** Data3 Type: ObjID Data 4:Null Data4 Type: ushort
NETO deleted object FIR only	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Net Fragment ID** Data3 Type: netfragID Data 4:Null Data4 Type: ushort
NETO bad fragment added	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Net Fragment ID** Data3 Type: netfragID Data 4: **Net Fragment ID** Data4 Type: netfragID
NETO fragment not found	NET[MCU]	Normal	Data3 could equal the NetObject State if we are talking about t NETxVerboseLog Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Net Fragment ID** Data3 Type: netfragID Data 4:Null Data4 Type: ushort
NETO object fragment sent	NET[MCU]	Normal	Data 1: **Net Fragment ID** Data1 Type: netfragID Data 2: **Net object ID** Data2 Type: ObjID Data 3: **Net object type** Data3 Type: RunnerType Data 4: **Data Length** Data4 Type: DataLength
NETO object fragment was declined	NET[MCU]	Normal	Data 1: **xmtresultcode** Data1 Type: xmtresultcode Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Net Fragment ID** Data3 Type: netfragID Data 4: **Data Length** Data4 Type: DataLength
NETO object list's timeout	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Fragment ID** Data2 Type: netfragID Data 3: **Net Object ID** Data3 Type: ObjID Data 4: **Net Object State** Data4 Type: ObjState

NETD ACK has failed	NET[MCU]	Normal	Data 1: **Destination RTU Address** Data1 Type: RTUAddress Data 2: **DataLength** Data2 Type: DataLength Data 3: **Result=(netdSendObject(DNPGetLocalAddress(DNPADD destAddr, DNPCIDNETOBJMGNT, datatosend, (NETOBJHEADDATALEN + 1)));** Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETD sending object	NET[MCU]	Normal	Data 1: **Destination RTU Address** Data1 Type: RTUAddress Data 2: **Data Length** Data2 Type: DataLength Data 3: **Result=DNPXmtMultiService** Data3 Type: ushort Data 4: **Connection ID** Data4 Type: connectID
NETD recieved object	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Net Object ID** Data2 Type: ObjID Data 3: **Net Object Type** Data3 Type: RunnerType Data 4: **Data Length** Data4 Type: DataLength
SW Phase A Overcurrent Start	ACW[MCU]	Normal	Current Over Threshold
SW Phase B Overcurrent Start	ACW[MCU]	Normal	Current Over Threshold
SW Phase C Overcurrent Start	ACW[MCU]	Normal	Current Over Threshold
SW Neutral Overcurrent Start	ACW[MCU]	Normal	Current Over Threshold
SW Phase A Overcurrent End	ACW[MCU]	Normal	Current Below Threshold Data 2: Data2 Type: Amps Data 3: Data3 Type: OCEndSeconds
SW Phase B Overcurrent End	ACW[MCU]	Normal	Current Below Threshold Data 2: Data2 Type: Amps Data 3: Data3 Type: OCEndSeconds
SW Phase C Overcurrent End	ACW[MCU]	Normal	Current Below Threshold Data 2: Data2 Type: Amps Data 3: Data3 Type: OCEndSeconds
SW Neutral Overcurrent End	ACW[MCU]	Normal	Current Below Threshold Data 2: Data2 Type: Amps Data 3: Data3 Type: OCEndSeconds
SW Phase A Fault Detected	ACW[MCU]	Normal	Current Over Threshold for Duration Threshold
SW Phase B Fault Detected	ACW[MCU]	Normal	Current Over Threshold for Duration Threshold
SW Phase C Fault Detected	ACW[MCU]	Normal	Current Over Threshold for Duration Threshold

SW Neutral Fault Detected	ACW[MCU]	Normal	Current Over Threshold for Duration THreshold
SW2 Phase A Overcurrent Start	ACW[MCU]	Normal	Current Over Threshold
SW2 Phase B Overcurrent Start	ACW[MCU]	Normal	Current Over Threshold
SW2 Phase C Overcurrent Start	ACW[MCU]	Normal	Current Over Threshold
SW2 Neutral Overcurrent Start	ACW[MCU]	Normal	Current Over Threshold
SW2 Phase A Overcurrent End	ACW[MCU]	Normal	Current Below Threshold Data 2: Data2 Type: Amps Data 3: Data3 Type: OCEndSeconds
SW2 Phase B Overcurrent End	ACW[MCU]	Normal	Current Below Threshold Data 2: Data2 Type: Amps Data 3: Data3 Type: OCEndSeconds
SW2 Phase C Overcurrent End	ACW[MCU]	Normal	Current Below Threshold Data 2: Data2 Type: Amps Data 3: Data3 Type: OCEndSeconds
SW2 Neutral Overcurrent End	ACW[MCU]	Normal	Current Below Threshold Data 2: Data2 Type: Amps Data 3: Data3 Type: OCEndSeconds
SW2 Phase A Fault Detected	ACW[MCU]	Normal	Current Over Threshold for Duration Threshold
SW2 Phase B Fault Detected	ACW[MCU]	Normal	Current Over Threshold for Duration Threshold
SW2 Phase C Fault Detected	ACW[MCU]	Normal	Current Over Threshold for Duration Threshold
SW2 Neutral Fault Detected	ACW[MCU]	Normal	Current Over Threshold for Duration Threshold
SW Phase A Loss of Voltage	ACW[MCU]	Normal	Voltage below Threshold
SW Phase A Normal Voltage	ACW[MCU]	Normal	Voltage above Threshold
SW Phase B Loss of Voltage	ACW[MCU]	Normal	Voltage below Threshold
SW Phase B Normal Voltage	ACW[MCU]	Normal	Voltage above Threshold
SW Phase C Loss of Voltage	ACW[MCU]	Normal	Voltage below Threshold
SW Phase C Normal Voltage	ACW[MCU]	Normal	Voltage above Threshold
SW Phase D Loss of Voltage	ACW[MCU]	Normal	Voltage below Threshold
SW Phase D Normal Voltage	ACW[MCU]	Normal	Voltage above Threshold
SW Phase E Loss of Voltage	ACW[MCU]	Normal	Voltage below Threshold
SW Phase E Normal Voltage	ACW[MCU]	Normal	Voltage above Threshold
SW Phase F Loss of Voltage	ACW[MCU]	Normal	Voltage below Threshold
SW Phase F Normal Voltage	ACW[MCU]	Normal	Voltage above Threshold
Direction 1 Current Flow On	ACW[MCU]	Normal	
Direction 1 Current Flow Off	ACW[MCU]	Normal	
Direction 2 Current Flow On	ACW[MCU]	Normal	
Direction 2 Current Flow Off	ACW[MCU]	Normal	

Fail Override Active	CMD[MCU]	Normal	Fail Override is active
Fail Override OFF	CMD[MCU]	Normal	Fail Override Disabled
Command	CMD[MCU]	Normal	Command Received Data 1: **Command Code** Data1 Type: CommandCode Data 2: **Command Code** Data2 Type: CommandSource Data 4: **SCADA Command** Data4 Type: SCADACommand
External Command Rejected	CMD[MCU]	Normal	External Command Rejected Data 1: **Command Code** Data1 Type: CommandCode
External Command Overriden	CMD[MCU]	Normal	Switch Op Allowed by Fail Ovewrride Data 1: **Command Code** Data1 Type: CommandCode
Hot Line Tag Active	CMD[MCU]	Normal	Hot Line Tag is active
Hot Line Tag OFF	CMD[MCU]	Normal	Hot Line Tag Disabled
SW Hot Line Tag Active	CMD[MCU]	Normal	SW Hot Line Tag is active
SW Hot Line Tag OFF	CMD[MCU]	Normal	SW Hot Line Tag Disabled
SW2 Hot Line Tag Active	CMD[MCU]	Normal	SW2 Hot Line Tag is active
SW2 Hot Line Tag OFF	CMD[MCU]	Normal	SW2 Hot Line Tag Disabled
SW3 Hot Line Tag Active	CMD[MCU]	Normal	SW3 Hot Line Tag is active
SW3 Hot Line Tag OFF	CMD[MCU]	Normal	SW3 Hot Line Tag Disabled
Hot Line Tag Set Local	CMD[MCU]	Normal	Hot Line Tag Set by FP or WiFi
Hot Line Tag Set Remote	CMD[MCU]	Normal	Hot Line Tag Set by SCADA
SCADA Command	CMD[MCU]	Normal	SCADA Command Received Data 1: **Command Code** Data1 Type: SCADACommandCod Data 3: **SCADA Command** Data3 Type: SCADACommand
SW S-T-L On	CMD[MCU]	Normal	SW S-T-L On
SW S-T-L Off	CMD[MCU]	Normal	SW S-T-L Off
SW2 S-T-L On	CMD[MCU]	Normal	SW2 S-T-L On
SW2 S-T-L Off	CMD[MCU]	Normal	SW2 S-T-L Off
Command Backup Issue	CMD[MCU]	Normal	Command Backup Issue Data 1: **Reason** Data1 Type: ushort Data 2: **Step** Data2 Type: ushort
Command Restore Issue	CMD[MCU]	Normal	Command Restore Issue Data 1: **Reason** Data1 Type: ushort Data 2: **Step** Data2 Type: ushort
Command Backup Success	CMD[MCU]	Normal	Command Backup Success
Command Restore Success	CMD[MCU]	Normal	Command Restore Success

SW Op in progress; cmd ignored	CMD[MCU]	Normal	SW Op in progress; cmd ignored Data 1: Data1 Type: CommandCode Data 2: Data2 Type: CommandSource
SW1 Open Operation	SWX[MCU]	Normal	SW1 Open
SW Not Open	SWX[MCU]	Normal	SW Close
SW2 Open Operation	SWX[MCU]	Normal	SW2 Open
SW2 Not Open	SWX[MCU]	Normal	SW2 Not Open
SW3 Open	SWX[MCU]	Normal	SW3 Open
SW3 Not Open	SWX[MCU]	Normal	SW3 Not Open
SW1 Closed	SWX[MCU]	Normal	SW1 Closed
SW Not Closed	SWX[MCU]	Normal	SW Not Closed
SW2 Closed	SWX[MCU]	Normal	SW2 Closed
SW2 Not Closed	SWX[MCU]	Normal	SW2 Not Closed
SW3 Closed	SWX[MCU]	Normal	SW3 Closed
SW3 Not Closed	SWX[MCU]	Normal	SW3 Not Closed
SW Disabled/Local Mode	SWX[MCU]	Normal	SW Has Errors or Manually Disabled
SW OK	SWX[MCU]	Normal	SW OK
SW2 Disabled/Local Mode	SWX[MCU]	Normal	SW2 Has Errors or Manually Disabled
SW2 OK	SWX[MCU]	Normal	SW2 OK
SW3 Disabled/Local Mode	SWX[MCU]	Normal	SW3 Has Errors or Manually Disabled
SW3 OK	SWX[MCU]	Normal	SW3 OK
SW Visible Disconnect Open	SWX[MCU]	Normal	SW Visible Disconnect Open
SW Visible Disconnect Closed	SWX[MCU]	Normal	SW Visible Disconnect Closed
SW2 Visible Disconnect Open	SWX[MCU]	Normal	SW2 Visible Disconnect Open
SW2 Visible Disconnect Closed	SWX[MCU]	Normal	SW2 Visible Disconnect Closed
SW Grounded	SWX[MCU]	Normal	SW Grounded
SW Not Grounded	SWX[MCU]	Normal	SW Not Grounded
SW2 Grounded	SWX[MCU]	Normal	SW2 Grounded
SW2 Not Grounded	SWX[MCU]	Normal	SW2 Not Grounded
SW Position Problem	SWX[MCU]	Normal	opensense == closesense
SW Position Good	SWX[MCU]	Normal	opensense != closesense
SW2 Position Problem	SWX[MCU]	Normal	opensense == closesense
SW2 Position Good	SWX[MCU]	Normal	opensense != closesense
SW3 Position Problem	SWX[MCU]	Normal	opensense == closesense
SW3 Position Good	SWX[MCU]	Normal	opensense != closesense
Low Pressure	SWX[MCU]	Normal	Low Pressure
Good Pressure	SWX[MCU]	Normal	Low Pressure Vanishes
Global Disable	SWX[MCU]	Normal	Global Disable

Global Disable OFF	SWX[MCU]	Normal	Global Disable switched OFF
Fault Interrupter Active	SWX[MCU]	Normal	Fault Interrupter Active
Fault Interrupter Cleared	SWX[MCU]	Normal	Fault Interrupter Cleared
Could not Reg Double Binary Status Point	SWX[MCU]	Normal	Could not Register Double Binary Status Point
Switch Operation Failed	SWX[MCU]	Normal	Switch Operation Failed Data 1: Data1 Type: SWnum Data 2: Data2 Type: SWOp
Sw Op Started	SWX[MCU]	Normal	Sw Op Started Data 1: Data1 Type: SWnum Data 2: Data2 Type: SWSWOp
WiFi Intrusion Attempt	WFM[MCU]	Normal	Alarms reported from WiFi module for replay attack or authen failure. Data 1:Debug Data Data1 Type: ushort
WiFi User Connected	WFM[MCU]	Normal	WiFi user connected.
WiFi User Disconnected	WFM[MCU]	Normal	WiFi user disconnected.
WiFi Intrusion Active	WFM[MCU]	Normal	WiFi intrusion active.
WiFi Intrusion Cleared	WFM[MCU]	Normal	WiFi intrusion cleared.
WiFi SCADA Disable	WFM[MCU]	Normal	WiFi disabled from SCADA.
WiFi SCADA Enable	WFM[MCU]	Normal	WiFi enabled from SCADA.
WiFi Test	WFM[MCU]	Normal	
GPS Enabled by IntelliLink	UTL[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort
GPS Disabled by IntelliLink	UTL[MCU]	Normal	Data 1:Debug Data Data1 Type: ushort
GPS Not Active Time Source On	UTL[MCU]	Normal	GPS is not the active time source set. Data 1: Data1 Type: TimeSource Data 2: Data2 Type: GPSStatus Data 3: Data3 Type: FixQuality
GPS Not Active Time Source Off	UTL[MCU]	Normal	GPS is not the active time source cleared. Data 1: Data1 Type: TimeSource Data 2: Data2 Type: GPSStatus Data 3: Data3 Type: FixQuality
GPS Status Changed	UTL[MCU]	Normal	GPS Status Changed. Data 1: Data1 Type: TimeSource

			Data 2: Data2 Type: GPSStatus Data 3: Data3 Type: FixQuality Data 4: Data4 Type: OneHexByte
Battery Low On	MOR[MCU]	Normal	
Battery Low Off	MOR[MCU]	Normal	
Battery Bad On	MOR[MCU]	Normal	
Battery Bad Off	MOR[MCU]	Normal	
Battery Test Hardware Error On	MOR[MCU]	Normal	
Battery Test Hardware Error Off	MOR[MCU]	Normal	
Battery Charger OverUnder Voltage On	MOR[MCU]	Normal	
Battery Charger OverUnder Voltage Off	MOR[MCU]	Normal	
OpenClose Contacts Not Mutually Excl On	MOR[MCU]	Normal	
OpenClose Contacts Not Mutually Excl Off	MOR[MCU]	Normal	
Internal Temperature Sensor Bad On	MOR[MCU]	Normal	
Internal Temperature Sensor Bad Off	MOR[MCU]	Normal	
External Temperature Sensor Bad On	MOR[MCU]	Normal	
External Temperature Sensor Bad Off	MOR[MCU]	Normal	
Limits of Travel Not Set On	MOR[MCU]	Normal	
Limits of Travel Not Set Off	MOR[MCU]	Normal	
Operator Decoupled From Switch On	MOR[MCU]	Normal	
Operator Decoupled From Switch Off	MOR[MCU]	Normal	
Motor Overload On	MOR[MCU]	Normal	
Motor Overload Off	MOR[MCU]	Normal	
Travel Fell Short On	MOR[MCU]	Normal	
Travel Fell Short Off	MOR[MCU]	Normal	
Overshot Position SP On Close On	MOR[MCU]	Normal	
Overshot Position SP On Close Off	MOR[MCU]	Normal	
Inspection Required On	MOR[MCU]	Normal	
Inspection Required Off	MOR[MCU]	Normal	
Calibration Required On	MOR[MCU]	Normal	
Calibration Required Off	MOR[MCU]	Normal	
Handle Not Stowed On	MOR[MCU]	Normal	
Handle Not Stowed Off	MOR[MCU]	Normal	
Switch Not Ready For High Speed Op On	MOR[MCU]	Normal	
Switch Not Ready For High Speed Op Off	MOR[MCU]	Normal	
Unrecoverable Error On	MOR[MCU]	Normal	

Unrecoverable Error Off	MOR[MCU]	Normal	
Motor Overload During Low Torque Op. On	MOR[MCU]	Normal	
Motor Overload During Low Torque Op Off	MOR[MCU]	Normal	
Motor Overload During High Trq Error On	MOR[MCU]	Normal	
Motor Overload During High Trq Error Off	MOR[MCU]	Normal	
Overshot Position SP On Open On	MOR[MCU]	Normal	
Overshot Position SP On Open Off	MOR[MCU]	Normal	
Secondary Position Error On	MOR[MCU]	Normal	
Secondary Position Error Off	MOR[MCU]	Normal	
Max Limits Spacing Exceeded On	MOR[MCU]	Normal	
Max Limits Spacing Exceeded Off	MOR[MCU]	Normal	
Variance Not Set Since Program Load On	MOR[MCU]	Normal	
Variance Not Set Since Program Load Off	MOR[MCU]	Normal	
Switch Not Ready For Override Op On	MOR[MCU]	Normal	
Switch Not Ready For Override Op Off	MOR[MCU]	Normal	
Internal Bad Command On	MOR[MCU]	Normal	
Internal Bad Command Off	MOR[MCU]	Normal	
Internal Bad State On	MOR[MCU]	Normal	
Internal Bad State Off	MOR[MCU]	Normal	
Internal Bad Encoder Checksum On	MOR[MCU]	Normal	
Internal Bad Encoder Checksum Off	MOR[MCU]	Normal	
Internal Bad Encoder Start Checksum On	MOR[MCU]	Normal	
Internal Bad Encoder Start Checksum Off	MOR[MCU]	Normal	
Internal Bad Encoder Range On	MOR[MCU]	Normal	
Internal Bad Encoder Range Off	MOR[MCU]	Normal	
Internal Bad Encoder Start Range On	MOR[MCU]	Normal	
Internal Bad Encoder Start Range Off	MOR[MCU]	Normal	
Internal Bad Encoder In Use On Reset On	MOR[MCU]	Normal	
Internal Bad Encoder In Use On Reset Off	MOR[MCU]	Normal	
Internal Bad Encoder Runaway On	MOR[MCU]	Normal	

Internal Bad Encoder Runaway Off	MOR[MCU]	Normal	
Internal Bad Encoder Drift On	MOR[MCU]	Normal	
Internal Bad Encoder Drift Off	MOR[MCU]	Normal	
Enc Sensor Detects Missing Encoder On	MOR[MCU]	Normal	
Enc Sensor Detects Missing Encoder Off	MOR[MCU]	Normal	
Control Power Not Present On	MOR[MCU]	Normal	
Control Power Not Present Off	MOR[MCU]	Normal	
Worn Switch On	MOR[MCU]	Normal	
Worn Switch Off	MOR[MCU]	Normal	
Knife Switch Not Engaged On	MOR[MCU]	Normal	
Knife Switch Not Engaged Off	MOR[MCU]	Normal	
Surge Suppressor Tripped On	MOR[MCU]	Normal	
Surge Suppressor Tripped Off	MOR[MCU]	Normal	
Battery Event	MOR[MCU]	Normal	Data 1: Data1 Type: BatteryEvent
Motor Event	MOR[MCU]	Normal	Data 1: Data1 Type: MotorOpReason Data 2: Data2 Type: MotorOperation
Switch Open On	MOR[MCU]	Normal	
Switch Open Off	MOR[MCU]	Normal	
Switch Closed On	MOR[MCU]	Normal	
Switch Closed Off	MOR[MCU]	Normal	
Switch Disabled On	MOR[MCU]	Normal	
Switch Disabled Off	MOR[MCU]	Normal	
Battery Status On	MOR[MCU]	Normal	
Battery Status Off	MOR[MCU]	Normal	
Maintenance Required On	MOR[MCU]	Normal	
Maintenance Required Off	MOR[MCU]	Normal	
Battery Test On	MOR[MCU]	Normal	
Battery Test Off	MOR[MCU]	Normal	
Cabinet Door Open On	MOR[MCU]	Normal	
Cabinet Door Open Off	MOR[MCU]	Normal	
Interlock Open On	MOR[MCU]	Normal	
Interlock Open Off	MOR[MCU]	Normal	
Prohibit Rest. Xmit Status Active	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort

			Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Enable Rest. Xmit Status Active	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Prohibit Rest. Xmit Status Clear	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Enable Rest. Xmit Status Clear	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Error Registering Team Peer	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
P2P message response	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Clr PR Status Control Pt Rcvd	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Clr Rest. Enab Stat Ctrl Pt Rcvd	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort

Prohibit Rest. Sent to Remote Dev	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Transfer Trip Sent to Remote Dev	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Error sending P.R. to Rmt Dev	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Error sending Xfer Trip to Rmt Dev	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Transfer Trip Enabled	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Transfer Trip Disabled	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Clearing Prohib Rest Blocked	RTL[MCU]	All	Clearing Prohibit Rest. Blocked Data 1:Debug Data Data1 Type: ushort Data 2:Debug Data Data2 Type: ushort Data 3:Debug Data Data3 Type: ushort Data 4:Debug Data Data4 Type: ClearPRBlocker
DNP3 Response received	RTL[MCU]	Normal	Data 1: Data1 Type: ushort

			Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
P.R. sent from local conditions	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Enable Restoration sent to RTU	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
RMT Xmit PR Enabd Local Conditions	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
RMT Xmit PR Disabd Local Conditions	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
RMT Xmit PR Enabd for SCADA	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
RMT Xmit PR Disabd for SCADA	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort
Clear PR on HLT removal disabled	RTL[MCU]	Normal	Data 1: Data1 Type: ushort Data 2: Data2 Type: ushort Data 3: Data3 Type: ushort Data 4: Data4 Type: ushort

DSP Event Overflow On	DIM[MCU]	Normal	
DSP Event Overflow Off	DIM[MCU]	Normal	
DSP Reset On	DIM[MCU]	Normal	
DSP Reset Off	DIM[MCU]	Normal	
DSP Configuration Error On	DIM[MCU]	Normal	
DSP Configuration Error Off	DIM[MCU]	Normal	
DSP Operation Suspend On	DIM[MCU]	Normal	
DSP Operation Suspend Off	DIM[MCU]	Normal	
DSP XBus Comm Error On	DIM[MCU]	Normal	
DSP XBus Comm Error Off	DIM[MCU]	Normal	
DSP Flash CRC Error On	DIM[MCU]	Normal	
DSP Flash CRC Error Off	DIM[MCU]	Normal	
DSP 7716 ADC Overflow On	DIM[MCU]	Normal	
DSP 7716 ADC Overflow Off	DIM[MCU]	Normal	
DSP Bad Address Latch On	DIM[MCU]	Normal	
DSP Bad Address Latch Off	DIM[MCU]	Normal	
DSP Not Running On	DIM[MCU]	Normal	
DSP Not Running Off	DIM[MCU]	Normal	
DSP In Diagnostic Mode On	DIM[MCU]	Normal	
DSP In Diagnostic Mode Off	DIM[MCU]	Normal	
NETX make runner source	NET[MCU]	Normal	Data 1: **Size of runner in RTU addresses** Data1 Type: RTUA Data 2: **Runner RTU lists** Data2 Type: RTUlists Data 3: **Runner node lists** Data3 Type: Nodelists Data 4:Null Data4 Type: ushort
NETX make runner RTUs	NET[MCU]	Normal	Data 1: **Size of runner in RTU addresses** Data1 Type: RTUA Data 2: **Runner RTU lists** Data2 Type: RTUlists Data 3: **Runner node lists** Data3 Type: Nodelists Data 4:Null Data4 Type: ushort
NETX make runner destination	NET[MCU]	Normal	Data 1: **Size of runner in RTU addresses** Data1 Type: RTUA Data 2: **Runner RTU lists** Data2 Type: RTUlists Data 3: **Runner node lists** Data3 Type: Nodelists Data 4:Null Data4 Type: ushort
NETX begin sendind activation runners	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort

			Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX activation object runner sent	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Data Length** Data3 Type: DataLength Data 4:Null Data4 Type: ushort
NETX forward activate runners	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Runner Index Number** Data2 Type: RunnerIndex Data 3: **Type of netlist distrubution[RunnerIndex]** Data3 Ty Data 4:Null Data4 Type: ushort
NETX forward activation runner index	NET[MCU]	Normal	Data 1: **Runner Index Number** Data1 Type: RunnerIndex Data 2: **Null** Data2 Type: ushort Data 3: **Null** Data3 Type: ushort Data 4: **Null** Data4 Type: ushort
NETX forward activation runner source	NET[MCU]	Normal	Data 1: **RTU Address** Data1 Type: RTUAddress Data 2: **Source Net Node ID** Data2 Type: SrcNodeID Data 3: **Destination Net Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX forward activation runner sent ok	NET[MCU]	Normal	NETX forwarding activation runner sent ok Data 1: **Runner Index Number** Data1 Type: RunnerIndex Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Data Length** Data3 Type: DataLength Data 4:Null Data4 Type: ushort
NETX Feeder Net is activated	NET[MCU]	Normal	Data 1: **Our position in the Node Index Table** Data1 Type: N Data 2: **Feeder Net Ids in the array** Data2 Type: FeederNetII Data 3: **Runner Index** Data3 Type: RunnerIndex Data 4:Null Data4 Type: ushort
NETX Feeder Net added to division list	NET[MCU]	Normal	Data 1: **VM table Index** Data1 Type: VMIndex Data 2: **Feeder Net Ids in the array** Data2 Type: FeederNetII Data 3: **Feeder Net CRCs** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort

NETX accept activation runner	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Runner index** Data2 Type: RunnerIndex Data 3: **Type of netlist distribution** Data3 Type: FeederNetO Data 4: **Null** Data4 Type: ushort
NETX accept activation runner failed	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Identify Actual Runner** Data2 Type: RunnerNumber Data 3: **Null** Data3 Type: ushort Data 4: **Null** Data4 Type: ushort
NETX accept activation runner result	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Identify Actual Runner** Data2 Type: RunnerNumber Data 3: **Null** Data3 Type: ushort Data 4: **Null** Data4 Type: ushort
NETX Make Nodes start	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: FeederNetID Data 2: **Local RTU Address** Data2 Type: RTUAddress Data 3: **Number of associated payload elements** Data3 Type NumPayLoad Data 4:Null Data4 Type: ushort
NETX Make Nodes compelete	NET[MCU]	Normal	Data 1: **Number of associated payload elements** Data1 Type NumPayLoad Data 2: **Number of Node IDs in the list** Data2 Type: NumNo Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX add nodes start	NET[MCU]	Normal	Data 1: **Null** Data1 Type: ushort Data 2: **Team Member Number** Data2 Type: TeamRecord Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4: **Null** Data4 Type: ushort
NETX add node id	NET[MCU]	Normal	Data 1: **Node ID** Data1 Type: SrcNodeID Data 2: **Switch** Data2 Type: SWnum Data 3: **Side information related to the switch** Data3 Type: S Data 4: **Null** Data4 Type: ushort

NETX add a new node	NET[MCU]	Normal	Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Number of Node Index entries in the Node Table** D NumNodeIndex Data 3: **Number of Node IDs in list** Data3 Type: NumNodeII Data 4:Null Data4 Type: ushort
NETX report adjacent NetList start	NET[MCU]	Normal	Data 1: **Left Node Index** Data1 Type: NodeIndex Data 2:Null Data2 Type: ushort Data 3: **Feeder Net CRC of the left** Data3 Type: FeederNetC Data 4:Null Data4 Type: ushort
NETX report adjacent to runner side1	NET[MCU]	Normal	Data 1: **Tie delievers an adjacent NL to a Runner Source fo** Data1 Type: RunnerType Data 2: **Destination Feeder Net ID** Data2 Type: FeederNetID Data 3: **Destination Feeder Net CRC** Data3 Type: FeederNet Data 4:Null Data4 Type: ushort
NETX report adj netlist obj send to s1	NET[MCU]	Normal	NETX report adjacent netlist object send to side1 Data 1: **Number of teams** Data1 Type: NumOfTeams Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Length of Feeder Net description method** Data3 Typ Data 4:Null Data4 Type: ushort
NETX Adj fdrnet netlist propagation done	NET[MCU]	Normal	NETX Adj feedernet netlist propagation done Data 1: **Adjacent FeederNet's CRC** Data1 Type: Hexushort Data 2: **Returned netlist runners of the adj feedernet** Data2 NumRunners Data 3: **Netlist runners in the adj feedernet** Data3 Type: Nu Data 4: **Sent netlist runners of adj feedernet** Data4 Type: Nu
NETX send adj Feeder Net Runner's qty	NET[MCU]	Normal	NETX send adjacent Feeder Net Runner's qty Data 1: **This is the runner's source device** Data1 Type: Devi Data 2: **Runner Source row number in nnet** Data2 Type: NetViewRowNum Data 3: **Number of runners** Data3 Type: NumRunners Data 4:Null Data4 Type: ushort
NETX send adj FdrNet runner's FdrNet ID	NET[MCU]	Normal	NETX send adjacent Feeder Net runner's feedernetID Data 1: **Runner Index** Data1 Type: RunnerIndex Data 2: **Feeder Net ID** Data2 Type: FeederNetID

			Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
NETX send adj Feeder Net runner's type	NET[MCU]	Normal	NETX send adjacent Feeder Net runner's type Data 1: **Runner Type** Data1 Type: RunnerType Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
NETX send adj FNet runner's adj FNet ID	NET[MCU]	Normal	NETX send adjacent Feeder Net runnner's adj FeederNetID Data 1: **Adjecent Feeder Net ID** Data1 Type: FeederNetID Data 2: **Adjecent Feeder Net CRC** Data2 Type: FeederNetCI Data 3: **Number of teams** Data3 Type: NumOfTeams Data 4:Null Data4 Type: ushort
NETX send adj Feeder Net runner's rtu	NET[MCU]	Normal	NETX send adjacent Feeder Net runner's rtu Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Length of Feeder Net description method** Data3 Typ Data 4: **Null** Data4 Type: ushort
NETX request netlist ok	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Net destination RTU Address** Data2 Type: RTUAdd Data 3: **Length of Feeder Net description method** Data3 Typ Data 4: **Null** Data4 Type: ushort
NETX reply Feeder Net req: crc is bad	NET[MCU]	Normal	NETX reply Feeder Net request, crc is bad Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Feeder Net CRC** Data2 Type: FeederNetCRC Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
NETX reply request sent ok	NET[MCU]	Normal	Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Destination RTU address** Data2 Type: RTUAddress Data 3: **Data Length** Data3 Type: DataLength Data 4:Null Data4 Type: ushort
NETX reply Feeder Net request not found	NET[MCU]	Normal	Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Feeder Net CRC** Data2 Type: FeederNetCRC Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort

NETX requested Feeder Net is a duplicate	NET[MCU]	Normal	Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Incoming Feeder Net ID** Data2 Type: FeederNetID Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX requested Feeder Net good	NET[MCU]	Normal	Data 1: **Netx is out of memory** Data1 Type: ErrorCodeNETX Data 2: **NetxFindspace(incoming number of teams)** Data2 T NumOfTeams Data 3: **Incoming number of teams** Data3 Type: NumOfTear Data 4:Null Data4 Type: ushort
NETX save requested Feeder Net ID	NET[MCU]	Normal	Data 1: **Incoming Feeder Net ID** Data1 Type: FeederNetID Data 2: **Incoming Feeder Net CRC** Data2 Type: FeederNetC Data 3: **Incoming number of teams** Data3 Type: NumOfTear Data 4:Null Data4 Type: ushort
NETX reequested Feeder Net saved ok	NET[MCU]	Normal	Data 1: **Size of team array** Data1 Type: ushort Data 2: **Size of teamsp data** Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX forward Feeder Net runner	NET[MCU]	Normal	Data 1: **Identify Actual Runner** Data1 Type: RunnerNumber Data 2: **Runner Index** Data2 Type: RunnerIndex Data 3: **Type of netlist distribution[RunnerIndex]** Data3 Ty FeederNetObjType Data 4:Null Data4 Type: ushort
NETX's forward runner's index	NET[MCU]	Normal	Data 1: **Incoming Runner's index** Data1 Type: RunnerIndex Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX forward Feeder Net destination rtu	NET[MCU]	Normal	Data 1: **RTU Address** Data1 Type: RTUAddress Data 2: **Source Node ID** Data2 Type: SrcNodeID Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort

NETX forward Feeder Net rnr sent: OK	NET[MCU]	Normal	NETX forward FeederNet runner sent,ok Data 1: **Runner Index** Data1 Type: RunnerIndex Data 2: **RTU address** Data2 Type: RTUAddress Data 3: **Data Length** Data3 Type: DataLength Data 4:Null Data4 Type: ushort
NETX forward adjacent Feeder Net sent ok	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Length of Feeder Net description method** Data3 Typ Data 4:Null Data4 Type: ushort
NETX save adjacent Feeder Net runner cvd	NET[MCU]	Normal	Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Feeder Net CRC** Data2 Type: FeederNetCRC Data 3: **Number of teams** Data3 Type: NumOfTeams Data 4: **Null** Data4 Type: ushort
NETX save adjacent Feeder Net IDs	NET[MCU]	Normal	Data 1: **Adjecent Feeder Net ID** Data1 Type: FeederNetID Data 2: **Adjecent Feeder Net CRC** Data2 Type: FeederNetCI Data 3: **Number of teams** Data3 Type: NumOfTeams Data 4:Null Data4 Type: ushort
NETX save adjacent Feeder Net node ID	NET[MCU]	Normal	Data 1: **Feeder Net Node ID** Data1 Type: SrcNodeID Data 2: **Node Index** Data2 Type: NodeIndex Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX save adj FeederNet dup chk hit	NET[MCU]	Normal	NETX save adjacent Feeder Net duplicate check hit Data 1: **Feeder Net ID in the array** Data1 Type: FeederNetID Data 2: **Adjecent Feeder Net ID** Data2 Type: FeederNetID Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX save adj Feeder Net number of teams	NET[MCU]	Normal	NETX save adjacent Feeder Net number of teams Data 1: **Adjecent Number of Teams** Data1 Type: NumOfTea Data 2: **Starting Index in the array** Data2 Type: FeederNetIr Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX save adj FNet; our num of teams	NET[MCU]	Normal	NETX save adjacent Feeder Net, our num of teams Data 1: **Our number of teams** Data1 Type: NumOfTeams

			Data 2: **Starting Index in the array** Data2 Type: FeederNetIr Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX adjacent FNet added to division net	NET[MCU]	Normal	NETX adjacent Feeder Net added to division net Data 1: **VM table index for this Feeder Net ID** Data1 Type: Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
NETX save adjacent Feeder Net conflict	NET[MCU]	Normal	Data 1: **Feeder Net ID in the array** Data1 Type: FeederNetID Data 2: **Adjecent Feeder Net ID** Data2 Type: FeederNetID Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX save adj FeederNet node indx failed	NET[MCU]	Normal	NETX save adjacent Feeder Net node index failed Data 1: **Node Index** Data1 Type: NodeIndex Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
NETX save Feeder Net runner recieved	NET[MCU]	Normal	Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Feeder Net CRC** Data2 Type: FeederNetCRC Data 3: **Number of teams** Data3 Type: NumOfTeams Data 4:Null Data4 Type: ushort
NETX save Feeder Net runner's nodeindex	NET[MCU]	Normal	Data 1: **Incoming Feeder Net ID** Data1 Type: FeederNetID Data 2: **Incoming Feeder Net CRC** Data2 Type: FeederNetC Data 3: **Node Index** Data3 Type: NodeIndex Data 4:Null Data4 Type: ushort
NETX save Feeder Net's dup replaced	NET[MCU]	Normal	NETX save Feeder Net's duplicate replaced Data 1: **Feeder Net ID in the array** Data1 Type: FeederNetID Data 2: **Incoming Feeder Net ID** Data2 Type: FeederNetID Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX save Feeder Net's present status	NET[MCU]	Normal	Data 1: **Feeder Net ID in the array** Data1 Type: FeederNetID Data 2: **Incoming Feeder Net ID** Data2 Type: FeederNetID Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort

NETX save Feeder Net runner success	NET[MCU]	Normal	Data 1: **Node Index** Data1 Type: NodeIndex Data 2: **Left Node/Right Node Index** Data2 Type: NodeIndex Data 3: **Incoming Feeder Net ID** Data3 Type: FeederNetID Data 4:Null Data4 Type: ushort
NETX save Feeder Net's both sides check	NET[MCU]	Normal	Data 1: **Incoming Node ID** Data1 Type: SrcNodeID Data 2: **Team Member Count** Data2 Type: ushort Data 3: **Node Index** Data3 Type: NodeIndex Data 4:Null Data4 Type: ushort
NETX save Feeder Net to our side	NET[MCU]	Normal	Data 1: **FeederNetID[NodeIndex]** Data1 Type: FeederNetID Data 2: **FeederNetID[otherIndex]** Data2 Type: FeederNetID Data 3: **Node Index** Data3 Type: NodeIndex Data 4:Null Data4 Type: ushort
NETX saved Feeder Net added to the index	NET[MCU]	Normal	Data 1: **Node Index** Data1 Type: NodeIndex Data 2: **Incoming Feeder Net ID** Data2 Type: FeederNetID Data 3: **Incoming Feeder Net CRC** Data3 Type: FeederNetC Data 4:Null Data4 Type: ushort
NETX saved FNet's nodeindx has failed	NET[MCU]	Normal	NETX saved Feeder Net's node index has failed Data 1: **Node Index** Data1 Type: NodeIndex Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4: **Null** Data4 Type: ushort
NETX accept the Feeder Net runner	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Identify actual runner** Data2 Type: RunnerNumber Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX register DNP Address List Size	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX communication check runner sent ok	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort

			Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Size of received runner in Bytes** Data3 Type: Runner Data 4:Null Data4 Type: ushort
NETX get communication chk runner number	NET[MCU]	Normal	NETX get communication check runner number Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2:Null Data2 Type: ushort Data 3: **Number of associated payload elements** Data3 Type: NumPayLoad Data 4:Null Data4 Type: ushort
NETX comm chk rnr's size is mismatched	NET[MCU]	Normal	NETX communication check runner's size is mismatched Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Object Length** Data2 Type: DataLength Data 3: **Size of recieved runner in Bytes** Data3 Type: Runner Data 4:Null Data4 Type: ushort
NETX get communication chk runner index	NET[MCU]	Normal	NETX get communication check runner index Data 1: **Size of recieved runner in Bytes** Data1 Type: Runner Data 2:Null Data2 Type: ushort Data 3: **Runner Index** Data3 Type: RunnerIndex Data 4:Null Data4 Type: ushort
NETX get communication chk rnr's rtu add	NET[MCU]	Normal	NETX get communication check runner's rtu address Data 1: **RTU Address** Data1 Type: RTUAddress Data 2: **Our local RTU Address** Data2 Type: RTUAddress Data 3: **Runner Index** Data3 Type: RunnerIndex Data 4:Null Data4 Type: ushort
NETX get comm chk rnr's dest rtu	NET[MCU]	Normal	NETX get communication check runner's destination rtu Data 1: **RTU Address** Data1 Type: RTUAddress Data 2: **Our local RTU Address** Data2 Type: RTUAddress Data 3: **Runner Index** Data3 Type: RunnerIndex Data 4:Null Data4 Type: ushort
NETX get comm chkrnr rnr internal node	NET[MCU]	Normal	NETX get communication checkrunner runner internal nodes Data 1: **RTU Address** Data1 Type: RTUAddress Data 2: **Runner Index** Data2 Type: RunnerIndex Data 3: **Node Count** Data3 Type: NodeCount Data 4:Null Data4 Type: ushort
NETX get comm chk runner rtu address	NET[MCU]	Normal	NETX get communication check runner rtu address Data 1: **RTU Address** Data1 Type: RTUAddress Data 2: **Our local RTU Address** Data2 Type: RTUAddress

			Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX get comm chk rnr indx fault	NET[MCU]	Normal	NETX get communication check runner index fault Data 1: **Size of the recived runner in Bytes** Data1 Type: Run Data 2:Null Data2 Type: ushort Data 3: **Runner Index** Data3 Type: RunnerIndex Data 4:Null Data4 Type: ushort
NETX begin forwarding comm chk rnr	NET[MCU]	Normal	NETX begin forwarding communication check runner Data 1: **Identify Actual Runner Number** Data1 Type: Runne Data 2:Null Data2 Type: ushort Data 3: **Size of runner in RTU addresses** Data3 Type: RTUA Data 4:Null Data4 Type: ushort
NETX forwarding comm check rnr sent ok	NET[MCU]	Normal	NETX forwarding communication check runner sent ok Data 1:Null Data1 Type: ushort Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Size of recived runner in Bytes** Data3 Type: RunnerS Data 4:Null Data4 Type: ushort
NETX accept communication check runner	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2:Null Data2 Type: ushort Data 3: **Size of runner in RTU Addresses** Data3 Type: RTUA Data 4:Null Data4 Type: ushort
NETX comm chk rnr size is mismatched	NET[MCU]	Normal	NETX communication check runner size is mismatched Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Data Length** Data2 Type: DataLength Data 3: **Size of recieved runner in Bytes** Data3 Type: Runner Data 4:Null Data4 Type: ushort
NETX communication check runner's index	NET[MCU]	Normal	Data 1: **Size of recieved runner in Bytes** Data1 Type: Runner Data 2:Null Data2 Type: ushort Data 3: **Null** Data3 Type: RunnerIndex Data 4:Null Data4 Type: ushort
NETX get comm check runner's data	NET[MCU]	Normal	NETX get communication check runner's data Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Object Length** Data2 Type: DataLength Data 3: **Size of runner in RTU Addresses** Data3 Type: RTUA Data 4:Null Data4 Type: ushort

NETX done accepting comm check runners	NET[MCU]	Normal	NETX done accepting communication check runners Data 1: **Claimed in dispatch** Data1 Type: EntryPointDevice Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Actual runner number from header** Data3 Type: Ru Data 4:Null Data4 Type: ushort
NETX make table of runner number	NET[MCU]	Normal	Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
NETX make runner table count	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3: **Runner Index** Data3 Type: RunnerIndex Data 4:Null Data4 Type: ushort
NETX acpt data collection rnr bad count	NET[MCU]	Normal	NETX accept data collection runner bad count Data 1: **Net Destination Node ID** Data1 Type: DestNodeID Data 2: **Runner Number** Data2 Type: RunnerNumber Data 3: **Runner's quantity** Data3 Type: RunnerQuantity Data 4:Null Data4 Type: ushort
NETX data delivery rnr node index failed	NET[MCU]	Normal	NETX data delivery Runner node index failed Data 1: **Node ID** Data1 Type: SrcNodeID Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX data runner information	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX data runner source rtu	NET[MCU]	Normal	Data 1: **The Normal State NetView length** Data1 Type: NET Data 2: **The local device number of the FN runner source d** Type: DeviceIndex Data 3: **RTU Address** Data3 Type: RTUAddress Data 4:Null Data4 Type: ushort
NETX data runner selection	NET[MCU]	Normal	

			Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Nnet row number** Data2 Type: NetViewRowNum Data 3: **The local device number of the FN runner source d** Type: DeviceIndex Data 4:Null Data4 Type: ushort
NETX peer is registered already	NET[MCU]	Normal	Data 1: **RTU Address** Data1 Type: RTUAddress Data 2:Null Data2 Type: ushort Data 3: **Registration Status** Data3 Type: RegistrationStatus Data 4:Null Data4 Type: ushort
NETX peer is deregistered; reregistering	NET[MCU]	Normal	NETX peer is deregistered, re-registering Data 1: **RTU Address** Data1 Type: RTUAddress Data 2:Null Data2 Type: ushort Data 3: **Registration Status** Data3 Type: RegistrationStatus Data 4:Null Data4 Type: ushort
NETX dest device node indx failure	NET[MCU]	Normal	NETX destination device node index failure Data 1: **Find this node in the NLTA arrays** Data1 Type: Ind Data 2: **VM table Index** Data2 Type: VMIndex Data 3: **Index into the substation list for Feeder Net "tree r** Type: Substation Data 4:Null Data4 Type: ushort
NETX dispatch designer Feeder Net object	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX dispatch FeederNet source and dest	NET[MCU]	Normal	NETX dispatch Feeder Net source and destination Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX dispatch Feeder Net dest Node ID	NET[MCU]	Normal	NETX dispatch Feeder Net destination Node ID Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Incoming Feeder Net ID** Data2 Type: FeederNetID Data 3: **Incoming Feeder Net CRC** Data3 Type: FeederNetC Data 4:Null Data4 Type: ushort
NETX dispatch Feeder Net delivery index	NET[MCU]	Normal	Data 1: **Incoming Feeder Net ID** Data1 Type: FeederNetID

			Data 2: **Incoming Feeder Net CRC** Data2 Type: FeederNetC Data 3: **Runner Index** Data3 Type: RunnerIndex Data 4:Null Data4 Type: ushort
NETX dispatch Feeder Net delivery save	NET[MCU]	Normal	Data 1: **Destination Index** Data1 Type: DestinationIndex Data 2: **Runner Number** Data2 Type: RunnerNumber Data 3: **Size of runner in RTU Addresses** Data3 Type: RTU Data 4:Null Data4 Type: ushort
NETX dispatch comm chk source and dest	NET[MCU]	Normal	NETX dispatch communication check source and destination Data 1: **Source RTU Adress** Data1 Type: RTUAddress Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX dispatch comm check FeederNet ID	NET[MCU]	Normal	NETX dispatch communication check Feeder Net ID Data 1: **Clamied in Dispatch** Data1 Type: EntryPointDevice Data 2: **Destination Index** Data2 Type: DestinationIndex Data 3: **Number of associated payload elements** Data3 Type NumPayLoad Data 4:Null Data4 Type: ushort
NETX dispatch comm check sent to	NET[MCU]	Normal	NETX dispatch communication check sent to Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
NETX dispatch comm check handled	NET[MCU]	Normal	NETX dispatch communication check handled Data 1: **Runner Index** Data1 Type: RunnerIndex Data 2: **Null** Data2 Type: ushort Data 3: **Null** Data3 Type: ushort Data 4: **Null** Data4 Type: ushort
NETX dispatch netlist rqst src and dest	NET[MCU]	Normal	NETX dispatch netlist request source and destination Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX dispatch netlist request FNet ID	NET[MCU]	Normal	NETX dispatch netlist request Feeder Net ID Data 1: **Destination RTU Address** Data1 Type: RTUAddress Data 2: **FeederNetID** Data2 Type: FeederNetID

			Data 3: **FeederNetCRC** Data3 Type: FeederNetCRC Data 4:Null Data4 Type: ushort
NETX dispatch primary rqst netlist resp	NET[MCU]	Normal	NETX dispatch primary request netlist response Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX dispatch prim netlist src and dest	NET[MCU]	Normal	NETX dispatch primary netlist source and destination Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX dispatch primary netlist source ID	NET[MCU]	Normal	Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Data Length** Data2 Type: DataLength Data 3: **Net source Node ID** Data3 Type: SrcNodeID Data 4:Null Data4 Type: ushort
NETX dispatch primary netlist indx found	NET[MCU]	Normal	NETX dispatch primary netlist index found Data 1: **Feeder Net ID** Data1 Type: FeederNetID Data 2: **Feeder Net CRC** Data2 Type: FeederNetCRC Data 3: **Runner Index** Data3 Type: RunnerIndex Data 4:Null Data4 Type: ushort
NETX dispatch primary netlist indx nums	NET[MCU]	Normal	NETX dispatch primary netlist indexed numbers Data 1: **Destination Index** Data1 Type: DestinationIndex Data 2: **Runner Number** Data2 Type: RunnerNumber Data 3: **Size of runner in RTU Addresses** Data3 Type: RTU Data 4:Null Data4 Type: ushort
NETX dispatch newadj delivery rnr src ST	NET[MCU]	Normal	NETX dispatch new adjacent delivery runner source ST Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Net Destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX dispatch new adj delivery rnr done	NET[MCU]	Normal	NETX dispatch new adjacent delivery runner done Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort

NETX dispatch adj netlist reporter src	NET[MCU]	Normal	NETX dispatch adjacent netlist reporter source Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Net destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX dispatch adj netlist report sending	NET[MCU]	Normal	NETX dispatch adjacent netlist reporter sending Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Number of actual data elments in DLV** Data2 Type: NumDataElements Data 3: **Net destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX dispat act Netlist rep fail to frwd	NET[MCU]	Normal	NETX dispatch Active Netlist reporter failed to forward Data 1: **Source RTU Address** Data1 Type: RTUAddress Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Net destination Node ID** Data3 Type: DestNodeID Data 4:Null Data4 Type: ushort
NETX state machine A state 01 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Waiting for IT Designer** Data2 Type: StatusFeedback Data 3: **Null** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machA state 01 depart FNet ID	NET[MCU]	Normal	NETX state machine A state 01 depart Feeder Net ID Data 1: **Incoming Feeder Net ID from IntelliTeam Designer** FeederNetID Data 2: **Incoming new Feeder Net CRC from IntelliTeam D** Data2 Type: FeederNetCRC Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX state machA state01 num of teams	NET[MCU]	Normal	NETX state machine A state 01 number of teams Data 1: **Feeder Net incoming team array number of teams** I INCNUMTEAM Data 2: **Number of Device-Wire pairs from prep function** D DeviceWirepairs Data 3: **Number of Incoming Feeder Net Flags from ITD** D FeederNetObjectFlags Data 4:Null Data4 Type: ushort
NETX state machine A state 02 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort

			Data 2: **Perform NETview Express analysis** Data2 Type: StatusFeedbackArea Data 3: **Waiting for IT Designer** Data3 Type: StatusFeedback Data 4:Null Data4 Type: ushort
NETX statemachA state02 arriv net stat	NET[MCU]	Normal	NETX state machine A state 02 arrival net status Data 1: **Number of devices in the Feeder Net** Data1 Type: NumDevicesinFN Data 2: **Number if entries in row,one for device,one for wir** Type: ExpressNetViewRow Data 3: **Number of paths in this Feeder Net** Data3 Type: Nu Data 4:Null Data4 Type: ushort
NETX state machine A state 03 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Create comm check runners and launch, then wait** Data2 Type: StatusFeedbackArea Data 3: **Perform NETview Express analysis** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
Register the peer RTU Address	NET[MCU]	Normal	Data 1: **Number of NETLIST runners in RunnerSource** Da NETLISTRNR Data 2: **Size of runner in RTU addresses** Data2 Type: RTUA Data 3: **RTU address of the controls for each stop along the** Type: RTUAddress Data 4:Null Data4 Type: ushort
NetX statemachA state03 peer regs fail	NET[MCU]	Normal	NetX state machine A state 03 peer registration has failed Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Registration Status** Data3 Type: RegistrationStatus Data 4:Null Data4 Type: ushort
NetX statemachA state03 all peers regs	NET[MCU]	Normal	NetX state machine A state 03 all peers registered Data 1: **Common check runners timeout counter** Data1 Typ Data 2:Null Data2 Type: ushort Data 3: **Number of NETLIST runners in RunnerSource** Da NETLISTRNR Data 4:Null Data4 Type: ushort
Netx state machA state 03 timeout failed	NET[MCU]	Normal	Netx state machine A state 03 timeout failed Data 1: **Common check of NETLIST runners in RunnerSou** Type: ushort

			Data 2:Null Data2 Type: ushort Data 3: **Number of NETLIST runners in RunnerSource** Da NETLISTRNR Data 4:Null Data4 Type: ushort
NETX statemachA state03 make rnrs failed	NET[MCU]	Normal	NETX state machine A state 03 make runners failed Data 1: **NETV source normal** Data1 Type: NETVSRC Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX state machine A state 04 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **go ahead and SEND the CCRs** Data2 Type: StatusFe Data 3: **Create comm check runners and launch, then wait** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine A state 04 is done	NET[MCU]	Normal	Data 1: **Common check of NETLIST runners in RunnerSou** Type: ushort Data 2:Null Data2 Type: ushort Data 3: **Number of NETLIST runners in RunnerSource** Da NETLISTRNR Data 4:Null Data4 Type: ushort
NETX state machine A state 05 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Start FN delivery Runners** Data2 Type: StatusFeedb Data 3: **go ahead and SEND the CCRs** Data3 Type: StatusFe Data 4:Null Data4 Type: ushort
NETX statemachA state 05 arrival FNet ID	NET[MCU]	Normal	NETX state machine A state 05 arrival FeederNET ID Data 1: **Feeder Net incoming team array number of teams** I INCNUMTEAM Data 2: **Number of NETLIST runners in RunnerSource** Da NETLISTRNR Data 3: **Incoming new Feeder Net ID from IntelliTeam Des** Type: FeederNetID Data 4:Null Data4 Type: ushort
NETX state machine A state 06 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Send activate runners** Data2 Type: StatusFeedbackA

			Data 3: **Start FN delivery Runners** Data3 Type: StatusFeedb Data 4:Null Data4 Type: ushort
NETX state machine A state 07 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Dashboard Feedback at the end of the successful de** Data2 Type: StatusFeedbackArea Data 3: **Send activate runners** Data3 Type: StatusFeedbackA Data 4:Null Data4 Type: ushort
NETX state machine A state 11 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Error Reporting** Data2 Type: StatusFeedbackArea Data 3: **Dashboard Feedback at the end of the successful de** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine C state 2 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Checks Registrations of the Destination RTU Addr** **CCRs to be sent** Data2 Type: StatusFeedbackArea Data 3: **Error Reporting** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX statemachC state2 regs is good	NET[MCU]	Normal	NETX state machine C state 2 registration is good Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Registration Status** Data3 Type: RegistrationStatus Data 4:Null Data4 Type: ushort
NETX state machC state 2 regs failed	NET[MCU]	Normal	NETX state machine C state 2 registration failed Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Registration Status** Data3 Type: RegistrationStatus Data 4:Null Data4 Type: ushort
NETX state machine C state 02 is done	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Current state of machine C State 2** Data2 Type: StatusFeedbackArea Data 3: **Previous state of machine C State 2** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine D state 01 Arrival	NET[MCU]	Normal	

			Data 1:Null Data1 Type: ushort Data 2: **Idle state waiting for management Sequence to be i** **upon need** Data2 Type: StatusFeedbackArea Data 3: **Null** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine D state 02 Arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Registers the RTU of the control that sent the Netli** **Runner to us** Data2 Type: StatusFeedbackArea Data 3: **Idle state waiting for management Sequence to be i** **upon need** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machD state 2 registered DNP	NET[MCU]	Normal	NETX state machine D state 2 registered DNP Data 1: **RTU Address** Data1 Type: RTUAddress Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX state machine D state 3 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Checks on and waits for successful registration of t** **Address from D0** Data2 Type: StatusFeedbackArea Data 3: **Registers the RTU of the control that sent the Netli** **Runner to us** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machD state 03 regs failed	NET[MCU]	Normal	NETX state machine D state 03 registration failed Data 1: **RTU Address** Data1 Type: RTUAddress Data 2:Null Data2 Type: ushort Data 3: **Registration Status** Data3 Type: RegistrationStatus Data 4:Null Data4 Type: ushort
NETX statemachD state 03 regs is good	NET[MCU]	Normal	NETX state machine D state 03 registration is good Data 1: **RTU Address** Data1 Type: RTUAddress Data 2:Null Data2 Type: ushort Data 3: **Registration Status** Data3 Type: RegistrationStatus Data 4:Null Data4 Type: ushort
NETX state machine D state 04 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Scans the list of reported Primary FeederNets and** **each one with delay between each request provided by nex** **waiting on the reply.** Data2 Type: StatusFeedbackArea

			Data 3: **Checks on and waits for successful registration of t** **Address from D0** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine D state 05 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Waits until the requested Feeder Net has been retu** **goes back to previous state** Data2 Type: StatusFeedbackArea Data 3: **Scans the list of reported Primary FeederNets and** **each one with delay between each request provided by nex** **waiting on the reply.** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine D state 06 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **the list of reported Adjacent FeederNets and reque** **with delay between each request provided by next state wa** **reply** Data2 Type: StatusFeedbackArea Data 3: **Waits until the requested Feeder Net has been retu** **goes back to previous state** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine D state 07 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Waits until the requested Feeder Net has been retu** **goes back to previous state** Data2 Type: StatusFeedbackArea Data 3: **The list of reported Adjacent FeederNets and requ** **with delay between each request provided by next state wa** **reply** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine D state 08 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **All Delivered now, proceed to apply** Data2 Type: StatusFeedbackArea Data 3: **Waits until the requested Feeder Net has been retu** **goes back to previous state** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machD state8 make div net	NET[MCU]	Normal	NETX state machine D state 8 make division net Data 1: **Record Where in VM find array this net list is in** D ushort Data 2: **Feeder Net ID** Data2 Type: FeederNetID Data 3: **Feeder Net CRC** Data3 Type: FeederNetCRC Data 4: **Null** Data4 Type: ushort

NETX state machine D state 9 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Registers New RTUs and copies our DivisionNet inf** **appropriate tables** Data2 Type: StatusFeedbackArea Data 3: **All Delivered now, proceed to apply** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine D state 09 Make Nodes	NET[MCU]	Normal	Data 1: **Index into the substation list for Feeder Net "tree r** Type: Substation Data 2: **Power Source Data-Starting Row in Present NetVie** **Power Source** Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX state machD state9 make nodes fail	NET[MCU]	Normal	NETX state machine D state 9 make nodes fail Data 1:Null Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX state machine D state 10 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Cleanup and return to Idle state** Data2 Type: StatusFeedbackArea Data 3: **All Delivered now, proceed to apply** Data3 Type: StatusFeedbackArea Data 4: **Null** Data4 Type: ushort
NETX state machine E state 01 arrival	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Runner Source Device Data Collection and Data Di** **manager IDLEStarts Runners on signal from State Machi** **when it finds we are a runner source** Data2 Type: StatusFee Data 3: **Cleanup and return to Idle state** Data3 Type: StatusFeedbackArea Data 4:Null Data4 Type: ushort
NETX state machine F state 01 Arrival	NET[MCU]	Normal	Data 1: **Runner Number for non-runner source devices** Dat RunnerNumber Data 2: **State Machine F Manages sequences of operations i** **multiple simultaneous, Netlist Distributions** Data2 Type:

			StatusFeedbackArea Data 3: **Null** Data3 Type: StatusFeedbackArea Data 4: **Null** Data4 Type: ushort
NETX state machine F state 02 Arrival	NET[MCU]	Normal	Data 1: **Runner Number for non-runner source devices** Dat RunnerNumber Data 2: **This state waits for the arrival of all required Activ Runners then applies the new Feeder Net to NETV and DA** Type: StatusFeedbackArea Data 3: **State Machine F Manages sequences of operations i multiple simultaneous, Netlist Distributions** Data3 Type: StatusFeedbackArea Data 4: **Null** Data4 Type: ushort
NETX state machine F state 03 Arrival	NET[MCU]	Normal	Data 1: **Runner Number for non-runner source devices** Dat RunnerNumber Data 2: **This state waits for the arrival of all required Activ Runners then applies the new Feeder Net to NETV and DA** Type: StatusFeedbackArea Data 3: **State Machine F Manages sequences of operations i multiple simultaneous, Netlist Distributions** Data3 Type: StatusFeedbackArea Data 4: **Null** Data4 Type: ushort
NETX state machine F state 04 Arrival	NET[MCU]	Normal	Data 1: **Runner Number for non-runner source devices** Dat RunnerNumber Data 2: **Releases our claim on PushingHoldoff, then waits f appropriate runners to have returned,then Analyzes to see runner source, lists Runners if yes, checks RSD Registratio** Type: StatusFeedbackArea Data 3: **State Machine F Manages sequences of operations i multiple simultaneous, Netlist Distributions** Data3 Type: StatusFeedbackArea Data 4: **Null** Data4 Type: ushort
NETX state machF state 04 runner source	NET[MCU]	Normal	NETX state machine F state 04 runner source Data 1: **Runner Source Device** Data1 Type: RSD Data 2: **Null** Data2 Type: ushort Data 3: **Null** Data3 Type: ushort Data 4: **Null** Data4 Type: ushort
NETX state machine G state 01 Arrival	NET[MCU]	Normal	

			Data 1: **Null** Data1 Type: ushort Data 2: **State Machine G Manages sequences of operations Sending of Feeder Net Deployment Runner Objects. State IdleState** Data2 Type: StatusFeedbackArea Data 3: **Null** Data3 Type: StatusFeedbackArea Data 4: **Null** Data4 Type: ushort
NETx state machine G state 01 deployment	NET[MCU]	Normal	Data 1: **Incoming number of teams** Data1 Type: INCNUMTI Data 2: **Number of netlist runners** Data2 Type: NETLISTRN Data 3: **Feeder Net ID from IntelliTeam Designer** Data3 Typ FeederNetID Data 4: **Null** Data4 Type: ushort
NETX state machine G state 02 Arrival	NET[MCU]	Normal	Data 1: **Null** Data1 Type: ushort Data 2: **G State 2 Creates and sends (store object) until the required number of runners has been sent** Data2 Type: StatusFeedbackArea Data 3: **State Machine G Manages sequences of operations Sending of Feeder Net Deployment Runner Objects** Data3 StatusFeedbackArea Data 4: **Null** Data4 Type: ushort
NETX state machG state02 store object	NET[MCU]	Normal	NETX state machine G state 02 store object Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **RTU Address** Data2 Type: RTUAddress Data 3: **Data Length** Data3 Type: DataLength Data 4:Null Data4 Type: ushort
NETX state machH state02 storeage ok	NET[MCU]	Normal	NETX state machine H state 02 storeage ok Data 1: **Runner Number** Data1 Type: RunnerNumber Data 2: **Destination RTU Address** Data2 Type: RTUAddress Data 3: **Null** Data3 Type: ushort Data 4: **Null** Data4 Type: ushort
NETX too many in Netview row	NET[MCU]	Normal	NETX too many in Netview row (would overflow) Data 1: **Current row number** Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX too many Netview columns	NET[MCU]	Normal	Data 1: **Current row number** Data1 Type: ushort Data 2:Null Data2 Type: ushort

			Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
Overflow the device instance table	NET[MCU]	Normal	Data 1: **Device Type** Data1 Type: dev Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX too many substations in Feeder Net	NET[MCU]	Normal	Would overflow Data 1: **Number of team record** Data1 Type: nt Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX too many netx devices	NET[MCU]	Normal	Data 1:Null Data1 Type: ushort Data 2: **Number of device connection present,counter,list le** Type: nsngl Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX too many netx columns	NET[MCU]	Normal	If Data4 is 4 then data 1 would be the deviceCountTotal Data 1:Null Data1 Type: ushort Data 2: **Number of device connection present,counter,list le** Type: nsngl Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX too many netx rows	NET[MCU]	Normal	If Data4 is 10 then data 1 would be the deviceCountTotal Data 1:Null Data1 Type: ushort Data 2: **Number of device connection present,counter,list le** Type: nsngl Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX too many NETV rows	NET[MCU]	Normal	Data 1: **Current row number** Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX too many DGs in Feeder Net	NET[MCU]	Normal	Data 1: **Current row number** Data1 Type: ushort Data 2:Null Data2 Type: ushort

			Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort
NETX found DG in Feeder Net	NET[MCU]	Normal	Data 1: **Current row number** Data1 Type: ushort Data 2:Null Data2 Type: ushort Data 3:Null Data3 Type: ushort Data 4:Null Data4 Type: ushort