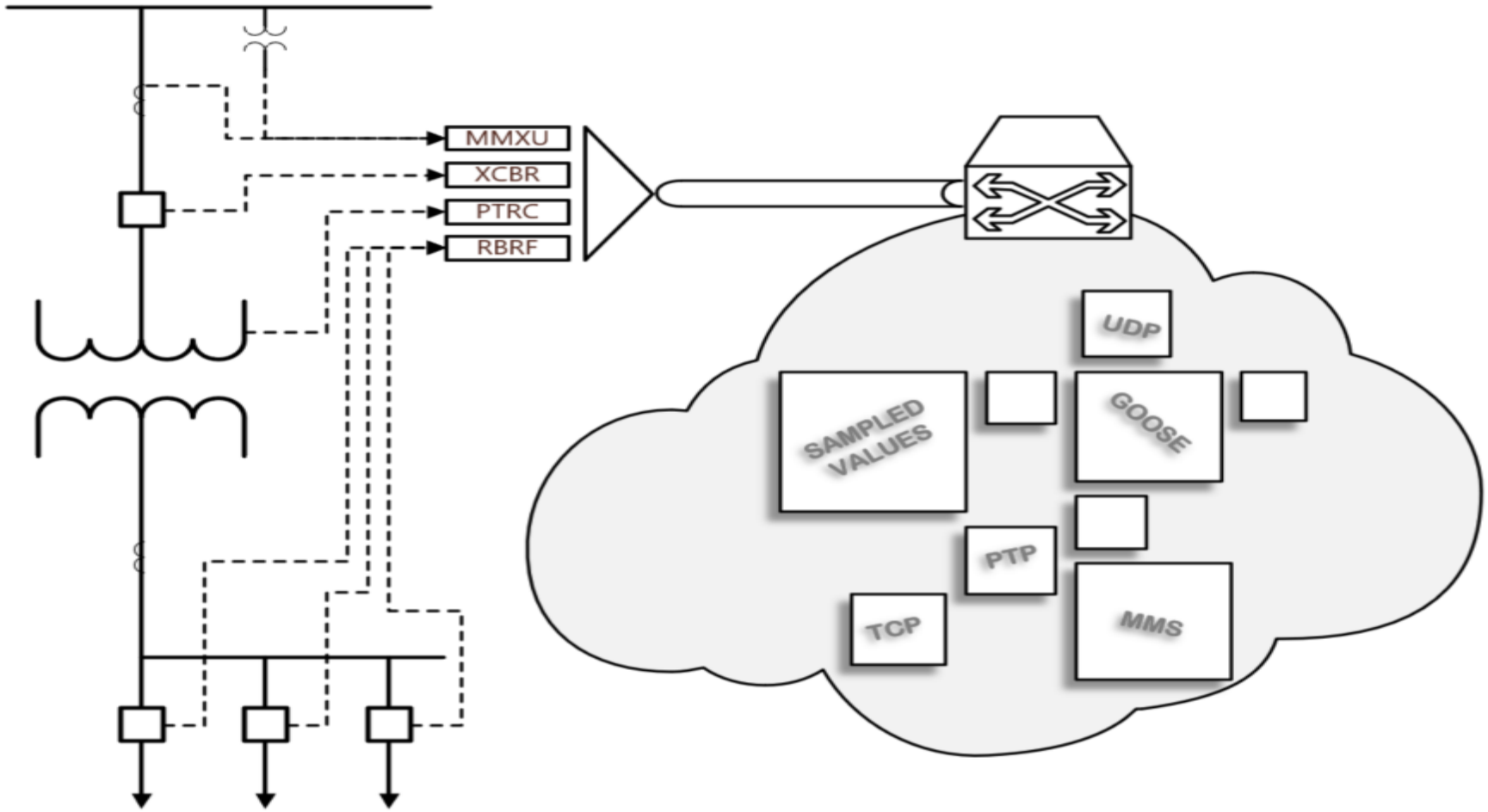


Communication bandwidth considerations for digital substations applications

Jake Groat, Galina S. Antonova, Benton Vandiver, Bharadwaj Vasudevan, Hitachi Energy

76th Annual Conference for Protective Relay Engineers, College Station, TX USA March 2023

Introduction

- Digital substations, instead of copper wiring, rely on the exchange of electrical signals represented as values in data communication streams

- Understanding communication bandwidth requirements and proper selection of communication technologies and equipment are critical in achieving required availability and reliability levels dictated by the protection and control applications

- Review and guidance on communication bandwidth utilization in digital substations is very timely, as the number of projects grows

Paper Content

- Communication Technologies in Digital Substations
  - IEC 61850 Sampled Values
  - IEC 61850 GOOSE messages
- Communication bandwidth calculations for
  - IEC 61850 Sampled Values
  - IEC 61850 GOOSE messages
- Traffic review from deployed projects and lab installations
  - IEC 61850 Sampled Values Captures
  - IEC 61850 GOOSE messages Captures
- Considerations on efficient bandwidth utilization
  - Handling background traffic and bursts of data
  - Selection of capable communication devices

Communication Technologies in Digital Substations

IEC 61850 Client-Server

- Point-to-point communication TCP/IP/Ethernet com for central monitoring and control
- Commands, reporting, logs, file transfer, etc

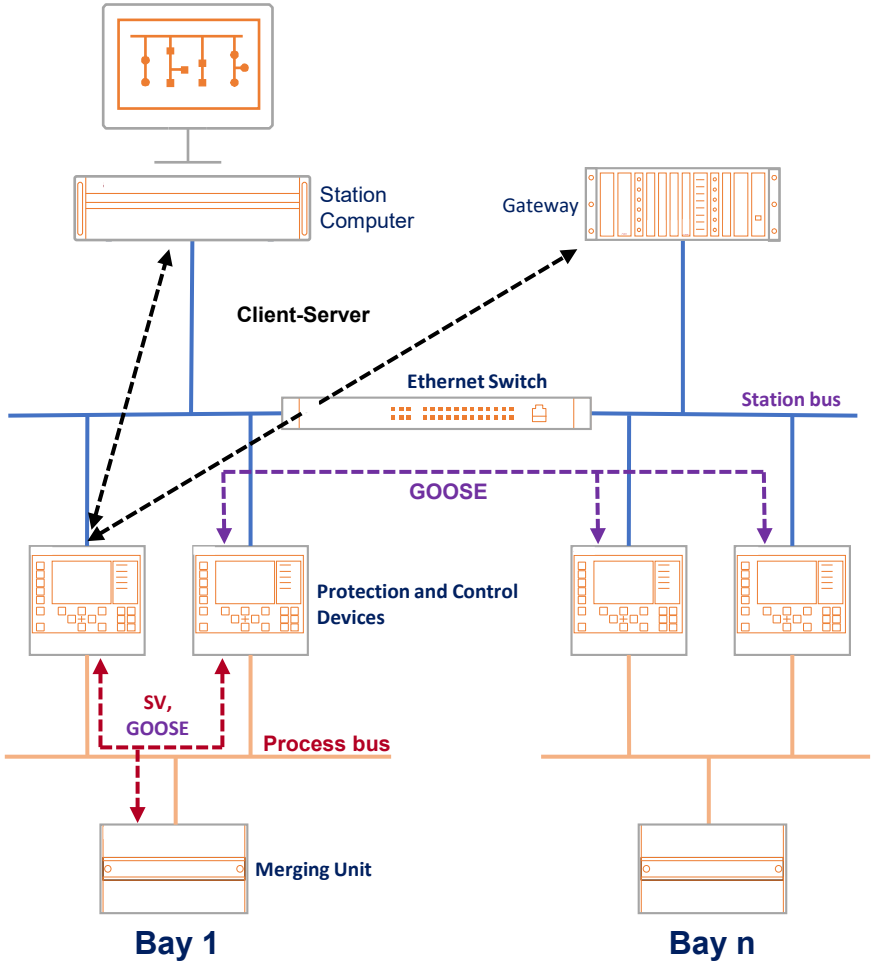
IEC 61850 GOOSE

- **Layer 2 Ethernet** real-time multicast com for station-wide monitoring, control, protection
- Binary data, indications, trip commands

IEC 61850 Sampled Values (SV)

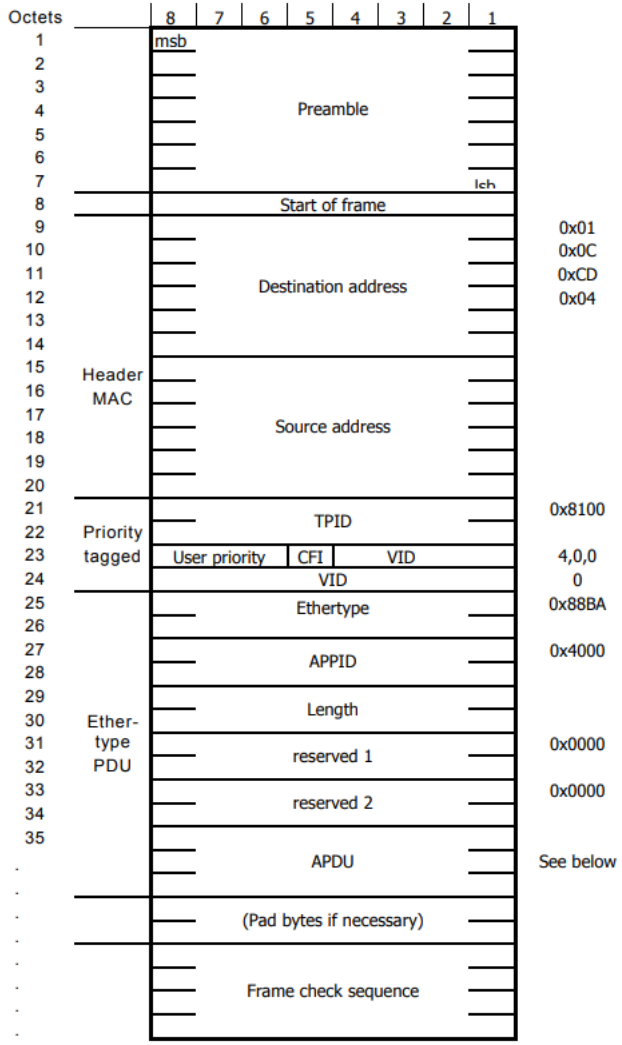
- **Layer 2 Ethernet** real data multicast com for analog measurements for control and protection
- Analog sampled values for currents and voltages

GOOSE = Generic Object-Oriented Substation Event



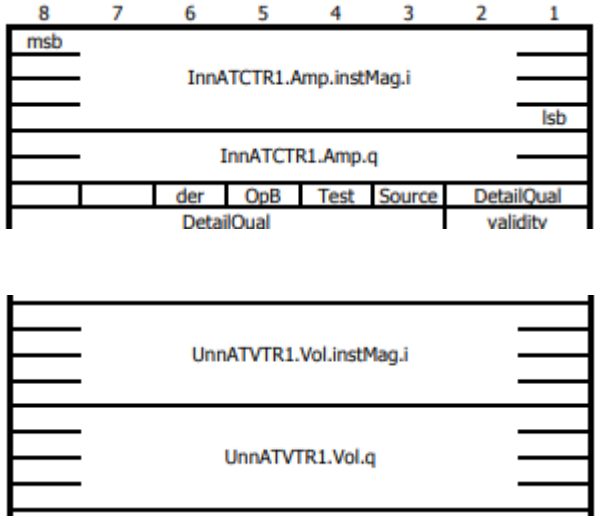
Sampled Values and GOOSE Ethernet Mapping

- Sampled Values and GOOSE data is mapped into Layer 2 Ethernet frames
- Data is wrapped by an Ethernet Header with
  - 6-byte Multicast Destination MAC Address
    MAC = 01-0C-CD-04-00-xx for SV
    MAC = 01-0C-CD-01-00-xx for GOOSE
  - 6-byte Source MAC Address
  - 4-byte IEEE 802.1Q Tag
  - 2-byte Ethernet type = 88-BA for SV
    = 88-B8 for GOOSE
  - 4-byte Frame Check sequence
- 7-byte Preamble and 1-byte Start of Frame are also present on the wire, so total wrapper is **23 bytes**



Sampled Values Data Set

- Sampled Values consist of 8 bytes
  - 4-byte magnitude value for current of voltage
  - 4-byte quality

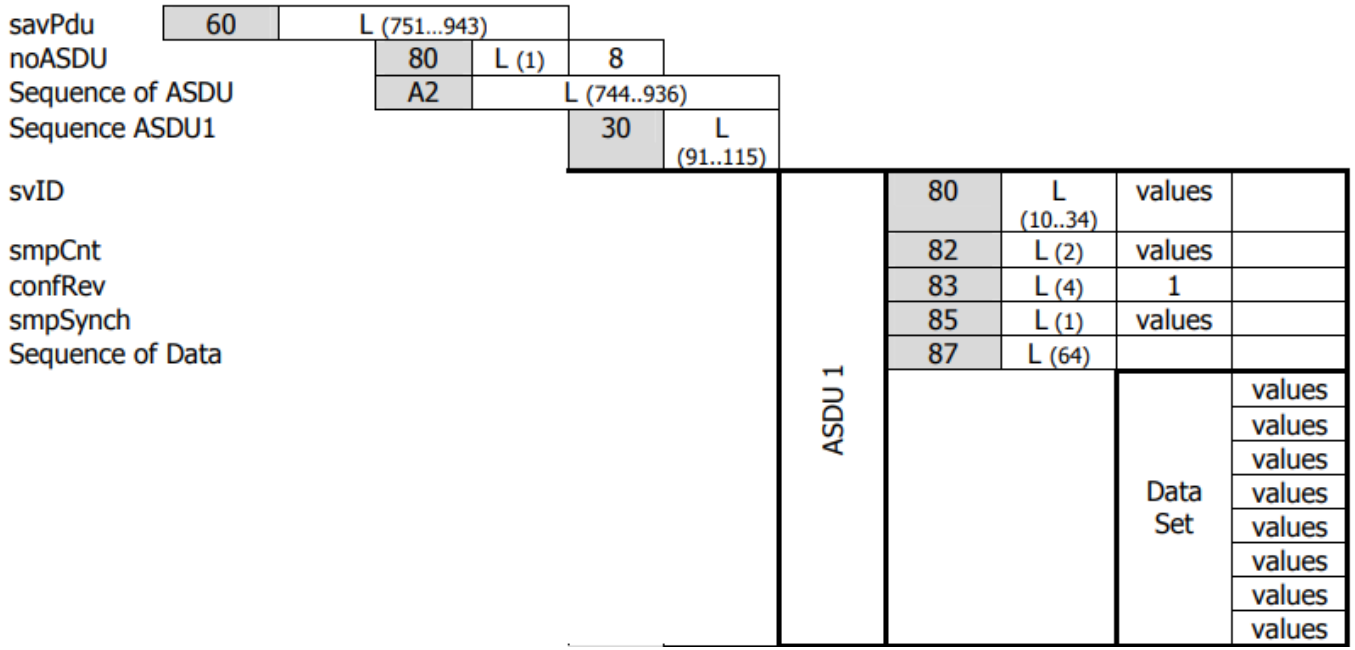


- IEC 61850-9-2LE Implementation Agreement defines transmission of
  - 4 current samples (3 phases and neutral) 32 bytes
  - 4 voltage samples (3 phases and neutral) 32 bytes

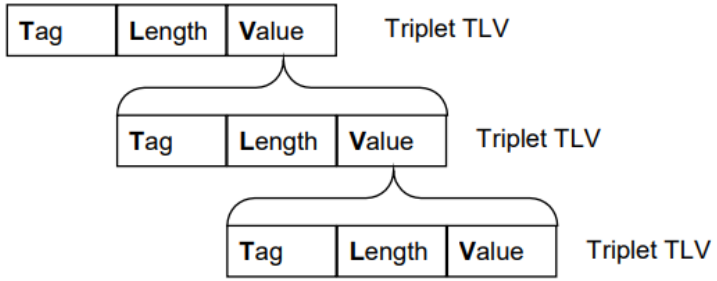
So, there are **64 bytes** per IEC 61850-9-2LE Data Set

Sampled Values Data Set Mapping

- Tag, Length, Value encoding is used, which adds additional **44 bytes**
  Resulting in **131-byte** SV frames on the wire, assuming 10-byte **svID**



svID can be 10 -34 bytes



Bandwidth Calculations for Sampled Values

- IEC 61850-9-2LE defines 2 sample rates:
  - **80 samples per power cycle** (mostly used)
  - 256 samples per power cycle
- For 80 samples for cycle of 60 Hz system is equal to **4800 samples/s**
- Each sample contains **131 byte** or **1048 bits** on the wire
- 1048 bits x 4800 samples/s = 5,030,400 bits/s = 4,912.5Kbits/s[1] ~ **4.9Mb/s**

- Thus, one (1) IEC 61850-9-2LE stream takes **4.9Mb/s**

  two (2) IEC 61850-9-2LE streams take **9.8Mb/s, etc.**

[1] Note that 1Kbit is equal to 1024 bits.

GOOSE Data Set

- GOOSE Data is fully configurable by the User (thus, data size is variable)
- Each data point
  - contains a defined for its data type number of bytes
  - may also include a 3-byte Quality (1 byte padding plus 2 byte quality bit string)
- Data encodings are defined by IEC 61850-8-1, Table A.1
- Like for SV, both data and quality are sent as Tag, Length, Value
- For GOOSE bandwidth calculation, let's use 1 Boolean value & Quality. Per Fixed-length mapping, it will occupy **8 bytes**:

1	2	3	4	5	6	7	8
83	01	FF	84	03			
Tag	Length	Value	Tag	Length	Value		

Tag = 0x83 identifies Boolean data, Tag = 0x84 is assigned for Quality

GOOSE Data Set Mapping

- GOOSE Header is specified by IEC 61850-8-1 Table A.2
- For Fixed-length encoding, it contains the following fields, Tags and Lengths

• goCBRef	0x80	Length determined by SCL configuration
• timeAllowedToLive	0x81	5
• datSet	0x82	Length determined by SCL configuration
• goID	0x83	Length determined by SCL configuration
• T	0x84	8
• stNum	0x85	5
• sqNum	0x86	5
• Simulation	0x87	1
• confRev	0x88	5
• ndsCom	0x89	1
• numDatSetEntries	0x8a	5

- For GOOSE bandwidth calculation, let's assume that variable-length fields is **10-bytes.** Then GOOSE Header will contain **87 bytes**

Note that each field is preceded by 1-byte Tag and 1-byte Length

Bandwidth Calculations for GOOSE

- Adding **23-bytes** of Ethernet wrapper to **8-byte** data and **87-byte** GOOSE Header, results in GOOSE frames containing **118 bytes** or **944 bits**.

- GOOSE messages have 2 configurable transmission times[1]
  - MaxTime a period for heartbeat-like transmissions, with no state changes
  - MinTime a period for transmissions when a state change occurs
- Assuming MinTime of 1ms, GOOSE peak transmission rate is 944,000[1] bits/s or 921.875 Kbits/s **~ 922 Kbits/s**
- Assuming MaxTime of 10s, GOOSE background transmission rate is **94.4 bits/s**

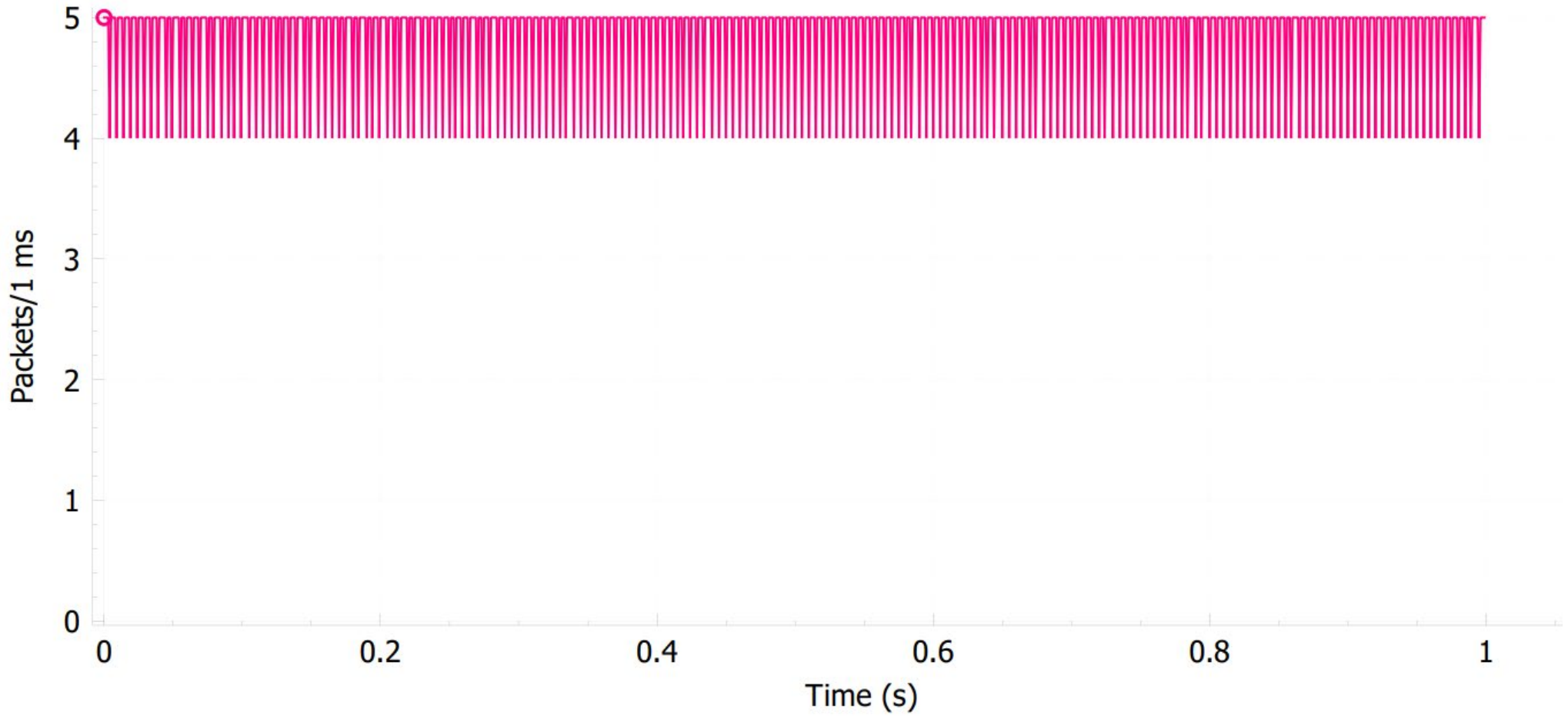
- Thus, with the above assumptions

  GOOSE background traffic with no state changes only takes **94.4 bits/s**

  GOOSE traffic during state changes takes ~ **922 Kbits/s**

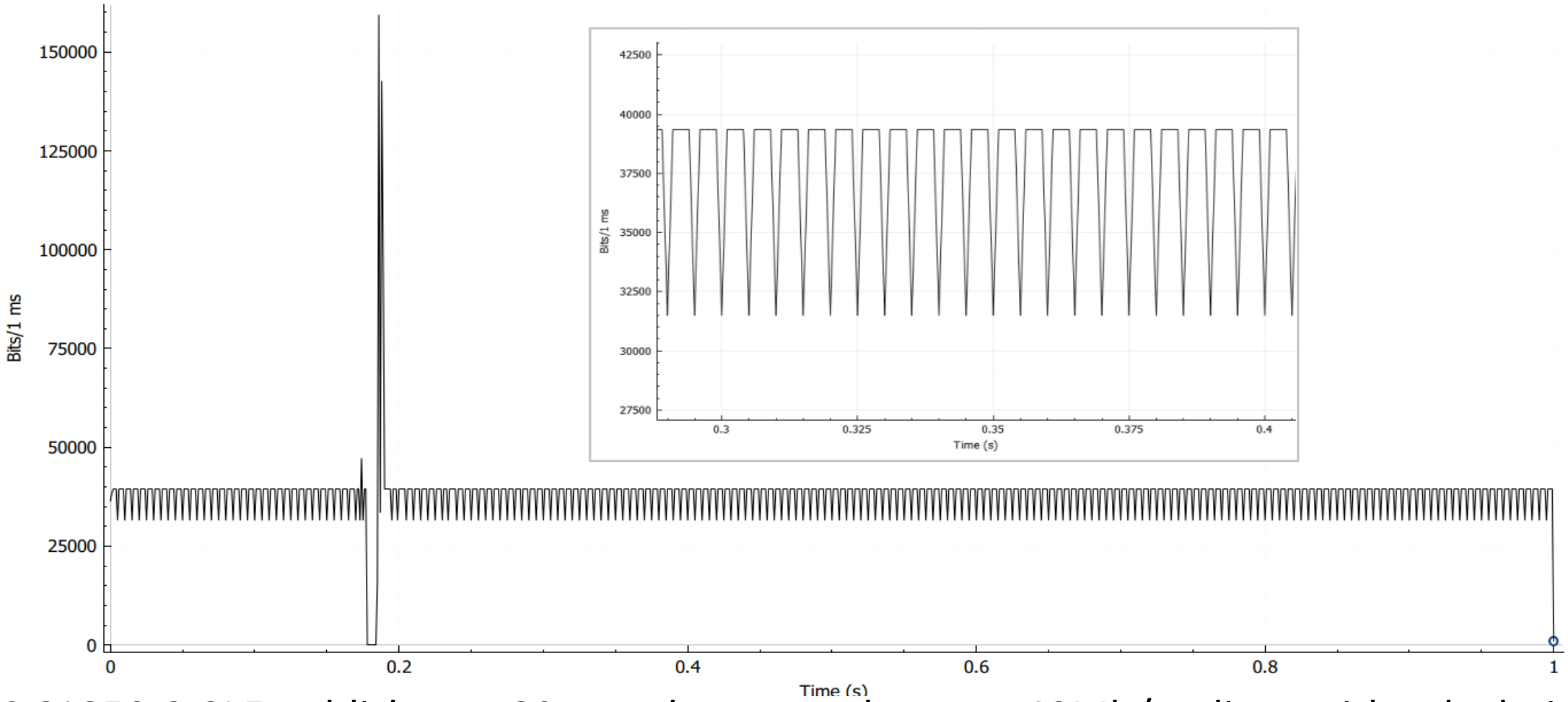
[1] Mechanisms to increase period from MinTime to MaxTime are implementation-specific

Traffic Capture 1: Single 9-2LE stream, 80 s/cycle



One IEC 61850-9-2LE publisher at 80 samples per cycle uses ~5Mb/s, same as calculated

Traffic Capture 2: 8 Sampled Values Streams



8 IEC 61850-9-2LE publisher at 80 samples per cycle uses ~40Mb/s, aligns with calculations

GOOSE Data Set

- GOOSE data consists of protection, control, status and alarm digitals
- Breaker status (DPS): 2-bits
- Size of the dataset is variable: Depends on application
- Less is more

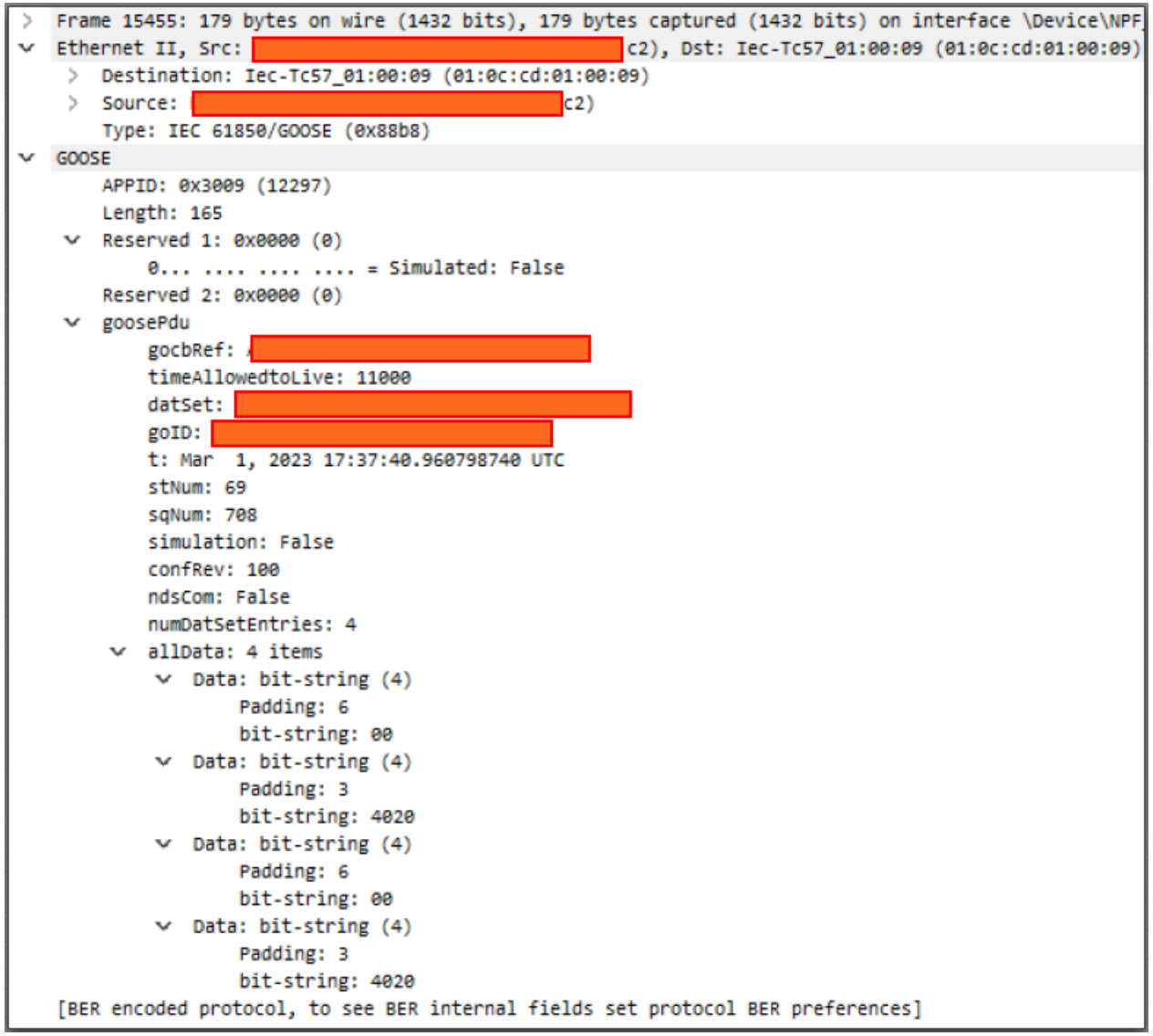
```
<DataSet name="PRO_Goose" desc="Protection GOOSE">
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind1" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind1" daName="q" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind2" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind2" daName="q" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind3" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind3" daName="q" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind4" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind4" daName="q" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind5" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind5" daName="q" fc="ST" />
    <FCDA ldInst="CTRL" prefix="SMP" lnClass="PTRC" lnInst="1" doName="Op" daName="general" fc="ST" />
    <FCDA ldInst="CTRL" prefix="SMP" lnClass="PTRC" lnInst="1" doName="Op" daName="q" fc="ST" />
    <FCDA ldInst="PROT" prefix="CC" lnClass="RBRF" lnInst="1" doName="OpEx" daName="general" fc="ST" />
    <FCDA ldInst="PROT" prefix="CC" lnClass="RBRF" lnInst="1" doName="OpEx" daName="q" fc="ST" />
</DataSet>
```

```
<DataSet name="GOOSE_Test" desc="GOOSE Test Dataset">
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind1" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind1" daName="q" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind2" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind2" daName="q" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind3" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind3" daName="q" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind4" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind4" daName="q" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind5" daName="stVal" fc="ST" />
    <FCDA ldInst="MON" prefix="SP16" lnClass="GAPC" lnInst="1" doName="Ind5" daName="q" fc="ST" />
    <FCDA ldInst="CTRL" prefix="SMP" lnClass="PTRC" lnInst="1" doName="Op" daName="general" fc="ST" />
    <FCDA ldInst="CTRL" prefix="SMP" lnClass="PTRC" lnInst="1" doName="Op" daName="q" fc="ST" />
    <FCDA ldInst="CTRL" prefix="SMP" lnClass="PTRC" lnInst="1" doName="Str" daName="general" fc="ST" />
    <FCDA ldInst="CTRL" prefix="SMP" lnClass="PTRC" lnInst="1" doName="Str" daName="q" fc="ST" />
    <FCDA ldInst="PROT" prefix="CC" lnClass="RBRF" lnInst="1" doName="OpEx" daName="general" fc="ST" />
    <FCDA ldInst="PROT" prefix="CC" lnClass="RBRF" lnInst="1" doName="OpEx" daName="q" fc="ST" />
    <FCDA ldInst="PROT" prefix="CC" lnClass="RBRF" lnInst="1" doName="OpIn" daName="general" fc="ST" />
    <FCDA ldInst="PROT" prefix="CC" lnClass="RBRF" lnInst="1" doName="OpIn" daName="q" fc="ST" />
</DataSet>
```

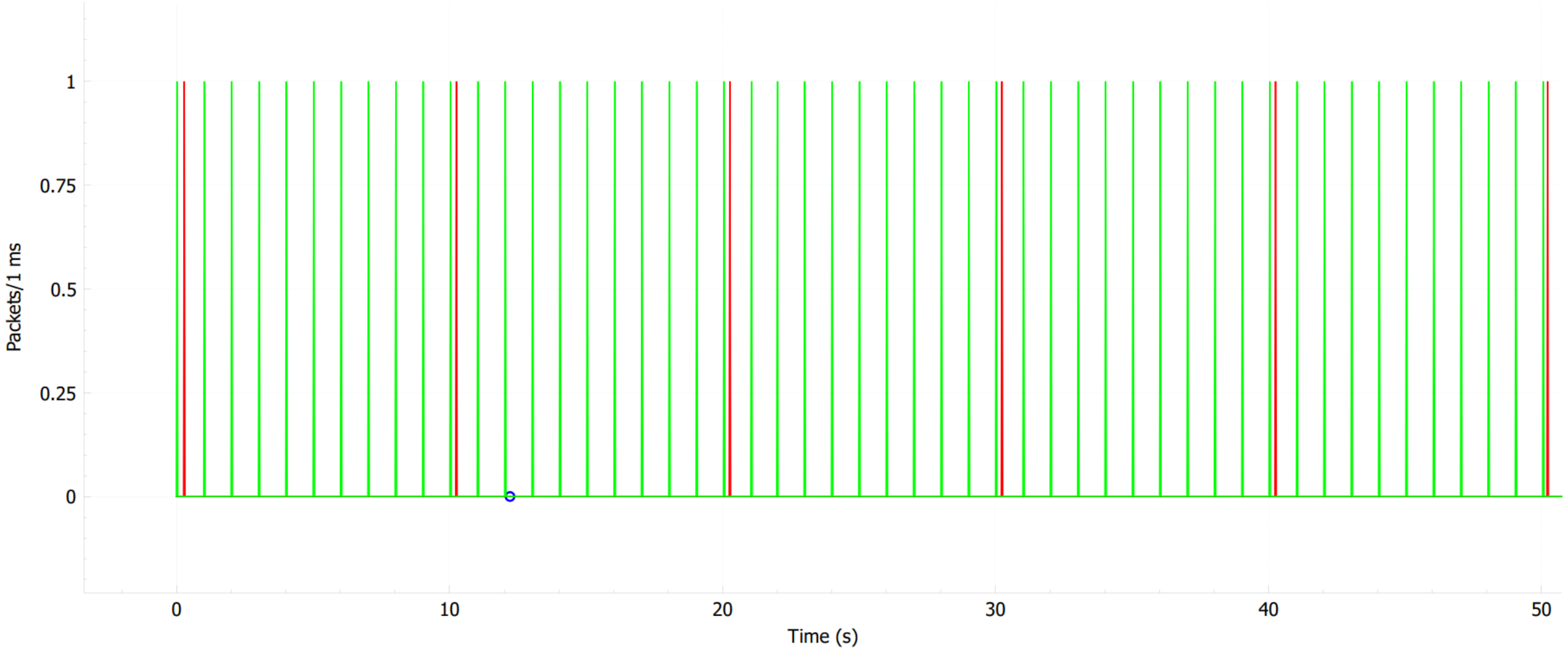
```
<DataSet name="Bkr_POS" desc="Breaker Status">
    <FCDA ldInst="CTRL" prefix="S" lnClass="XCBR" lnInst="1" doName="Pos" daName="stVal" fc="ST" />
    <FCDA ldInst="CTRL" prefix="S" lnClass="XCBR" lnInst="1" doName="Pos" daName="q" fc="ST" />
</DataSet>
```

GOOSE Data Model

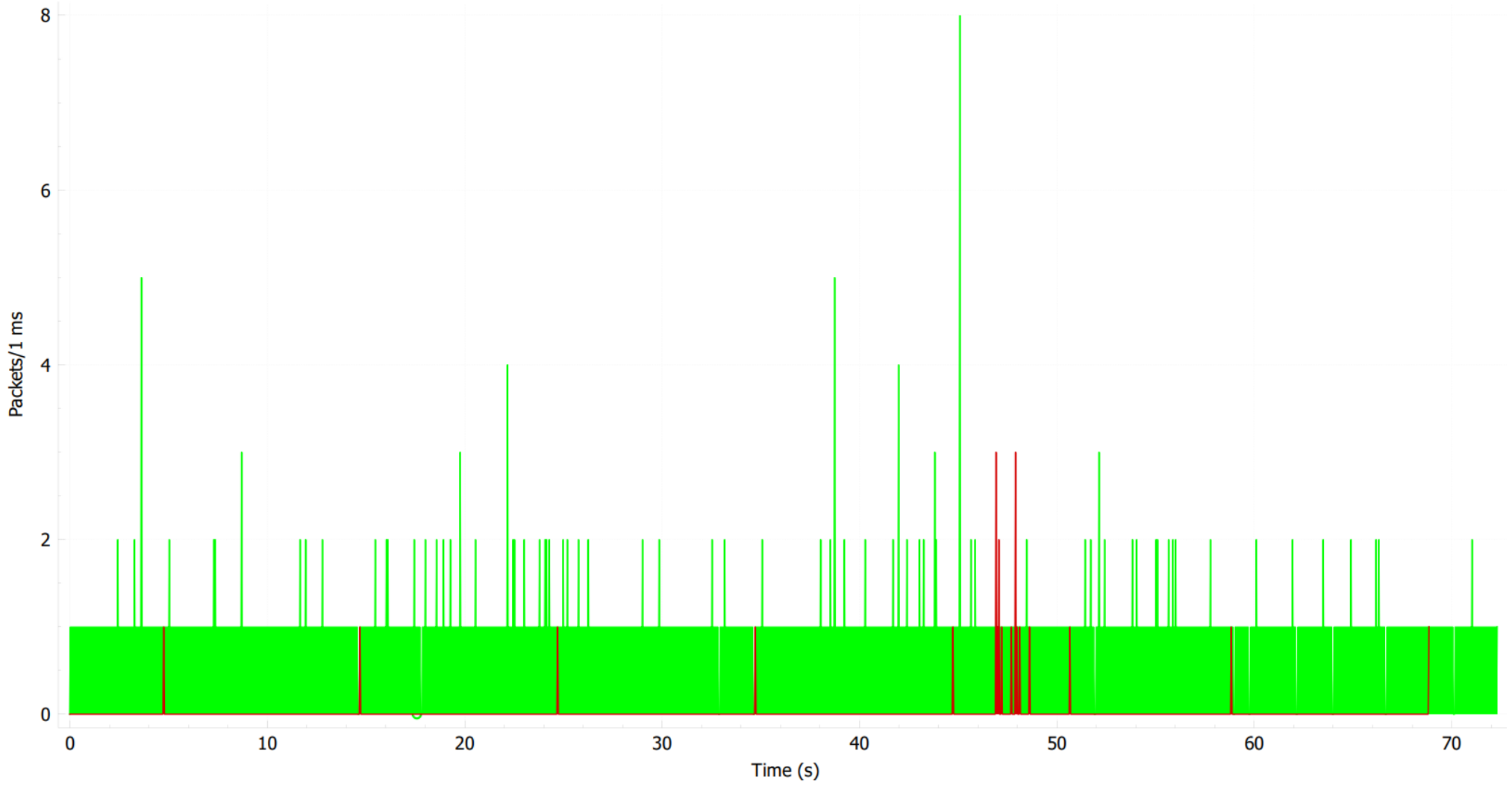
- Logical Device
  - ➢CTRL
- Logical Node
  - ➢XCBR
- Data Object
  - ➢Pos
- Data Attribute
  - ➢stVal



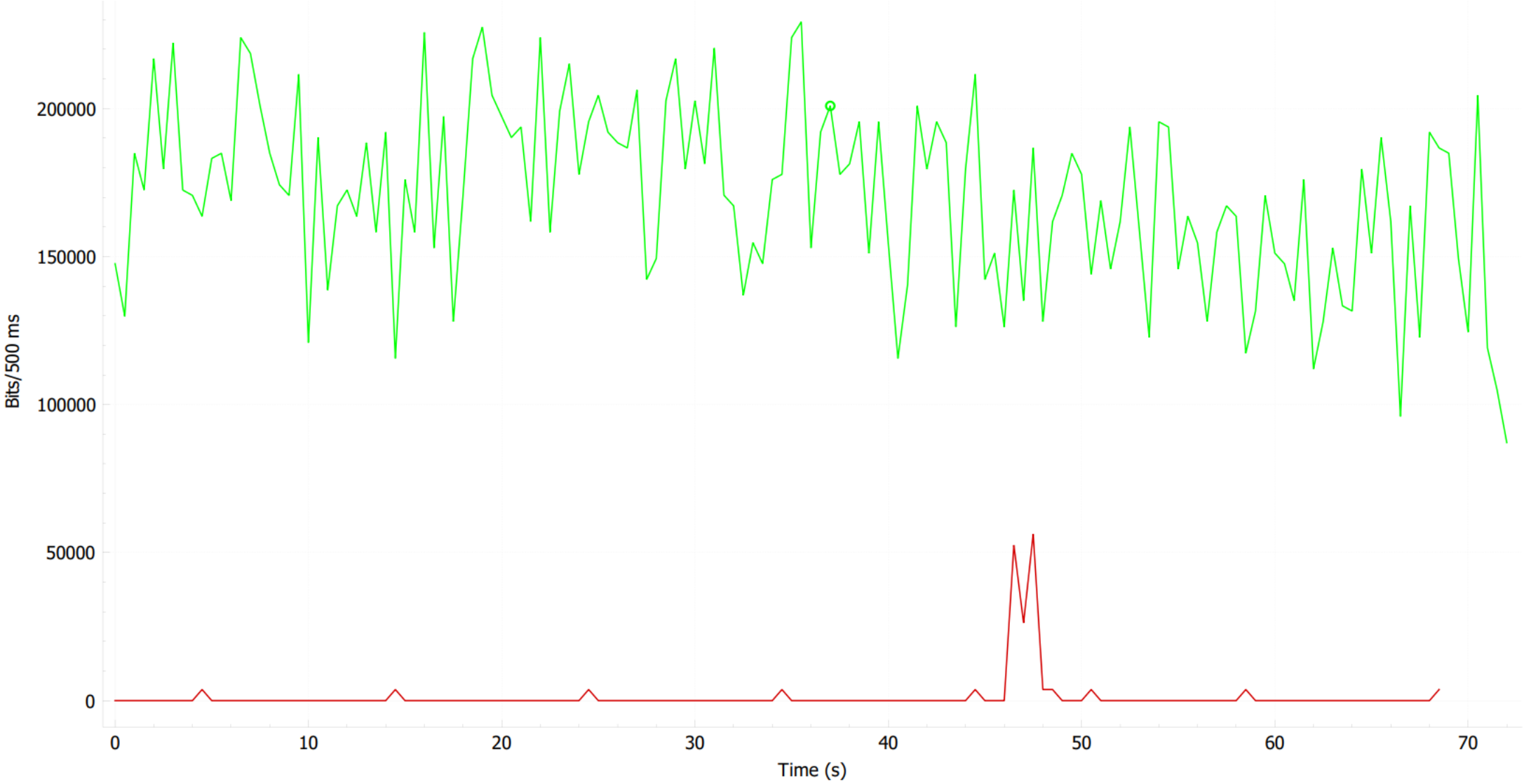
Traffic Capture 3: Two GOOSE publishers



Traffic Capture 4: GOOSE State Change



Traffic Capture 5: Bandwidth on State Change



Bandwidth Calculations for GOOSE Captures

- 2 different frames every 10 sec from Publisher 1, 2 GOOSE messages
  - Long frame has 469 bytes, and short frame has 202 bytes
  - Steady state transmission: (469 x 8) + (202 x 8) / 10sec= **536.8 bits/s**
  - Event Transmission: over 50 frames in 5 seconds
  - Initial event: ~30 short frames in 1 sec: 202 x 30 x 8 = **48,480 bits/s**

- 400 frames per sec from Publisher 2, 2 GOOSE messages
  - ~220 bytes per frame: 220 bytes x 8 bits x 400 s^{-1} = **704,000 bits/s**
  - Event transmission is indistinguishable from steady state traffic

Considerations on efficient bandwidth utilization

- Once communication bandwidth requirement was assessed and met, utilization of the available bandwidth can be improved.
- For stable background traffic pattern, IEDs with regular evenly spaced data transmissions should be selected
- Communication devices (Ethernet switches) selected should be capable of handling both background and expected during events burst traffic
- Communication devices should also have robust forwarding that minimizes unnecessary data broadcasting for MAC address learning
- Communication devices can also be configured for multicast filtering, rate limiting, as well as VLANs for traffic segregation

Conclusions

- Knowing theoretically and practically how much bandwidth is actually used for data exchange in normal conditions and during a state change is an important consideration for proper design and optimization of Ethernet communication infrastructure, including selection of suitable com devices
- Bandwidth utilization by SV is 4.9Mbits/s per IEC 61850-9-2LE stream
- Bandwidth utilization by GOOSE depends on application, calculated example showed ~95bits/s in normal state and 922Kbits/s during an event
- To reduce bandwidth usage and use available bandwidth efficiently mindful GOOSE engineering, IED and communication device selections are the key.

Thank you !