The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

University of Cape Town

# Application of IEC61850 protocol to implement peer-to-peer communication on General Electric (GE) relays: Evaluation of IEC61850 Performance

A thesis submitted to the
UNIVERSITY OF CAPE TOWN
In fulfilment of the requirements for the Degree of
MASTER OF SCIENCE IN ENGINEERING
(ELECTRICAL ENGINEERING)

By

Matankiso Moerane
B.Sc. (Electrical Engineering), UCT

Department of Electrical Engineering
University of Cape Town
Rondebosch 7701,
Cape Town
SOUTH AFRICA

March 2013

University of Cape Town

# Declaration

I hereby:

a) offer the University free licence to reproduce the above thesis in whole or in part for the purpose of research;

b) declare that:

i) The above thesis is my own work, from the beginning up to execution and that apart from the normal guidance from my supervisor, I have received no assistance apart from that stated below;

ii) To my knowledge neither the material or any part of the thesis material has been submitted in the past, or is submitted for the degree in the University or any other University, except that stated below,

iii) I am now presenting the thesis for examination of the Masters Degree

University of Cape Town

# Synopsis

Substation Automation (SA) and Distribution Automation (DA) have a significant role to play at electricity substation and distribution network levels. They have the potential to influence the overall performance of protecting devices in isolating, clearing faults in a network and in restoring supply to the customers.

The implementation of SA and DA is becoming a reality due to the use of intelligent electronic devices (IEDs). IEDs have the capability to perform protection, control and metering (PCM), and monitoring functions. Using IEDs, both SA and DA can monitor and control power delivery systems in substations and on the reclosers, reducing the duration and number of outages.

SA is said to be improving protection performance due to the Generic Object Oriented Substation Event (GOOSE) messages that can be rapidly transmitted in a network to perform protection and other functions. Based on the fact that the GOOSE messages only occupy three layers of the Open System Interconnection (OSI) model, in contrast to hardwired relay contacts which must traverse additional communication layers; the GOOSE messages are claimed to be faster than hardwired contacts on the relays. The objective of this research is to prove whether the GOOSE messages are indeed faster than the hardwired contacts.

The main questions that guide the thesis are, firstly, is the speed of GOOSE messages always faster than the speed of hard wired contacts of the relays? If not, what are the factors that can affect the performance of GOOSE messages? Secondly, is a non-conventional Breaker fail function using GOOSE messages possible, without using a conventional bus zone panel? Additionally, the research sought to establish the interoperability of IEDs from different vendors using the GOOSE message system. These questions are answered with the aid of laboratory experiments.

Regarding the first question, the results demonstrate that the GOOSE messages are faster than the physical contacts of the relays, but not always. The contributing factors affecting the performance of GOOSE messages are the debounce time of the contact inputs and the operating time of contact outputs of the IEDs. The shorter the combination of the debounce time and the operating time, the high the probability of GOOSE performance being slower than the hardwired contacts of the relays. In this research, where the combination of the debounce time and the operating time is 12 milliseconds or lower, the performance of GOOSE messages is slower than the hardwired contacts signals. This is a surprise as the performance of GOOSE messages are claimed to be faster than the hardwired contacts. Secondly, the non-conventional Breaker fail –bus strip operation was achieved without using a bus zone panel as an interface. GOOSE messages were successfully used to carry breaker fail signals from one GE relay/IED to another. Lastly, interoperability was implemented between the Omicron test set and the GE relay during the experiments in the lab. The GOOSE messages were sent from the F35 relays to the Omicron test set successfully, with the Omicron test set reacting as expected.

University of Cape Town

# Dedication

This thesis is dedicated to my mother Masabata Moerane, my husband David Mohlokoana and the extended family of Moerane for their love and loyal support.

# Acknowledgements

It is a pleasure to thank my supervisor Prof Komla Folly for his guidance and support during my Masters research.

I also thank the following people who contributed to my masters research:

- Tebogo Mohohlo from IST, thank you for being patient with all the questions I asked you.
- Petrus van Wyk from IST, thank you for your guidance with my masters project and for lending me your equipment.
- Christo Venter from Alectrix (Pty) Ltd, thank you for providing me with some of the software for me to be able to do my experiments, and for taking time to answer all my questions.
- Rethabile Melamu, thank you for all your advice and guidance. I truly appreciate everything you have done for me.
- Selby Mudau, thank you for all your support and advice. I truly appreciate your help.
- David Mohlokoana, for his encouragement and support.

Last but not least, I thank my God, my father, my everything. I thank Him for giving me strength and guidance.

University of Cape Town

# Table of Contents

University of Cape Town

University of Cape Town

# List of Figures

University of Cape Town

University of Cape Town

# List of Tables

University of Cape Town

# Acronyms

The major terms used in this report are listed below:

| | |
|---|---|
| **AEP** | American Electric Power |
| **ALS** | Advanced Loop Scheme |
| **ARC** | Auto- reclose |
| **APDU** | Application Protocol Data Unit |
| **ASDU** | Application Service Data Unit |
| **Broadcasting** | Transmitting messages to all nodes in a network |
| **CB** | Circuit Breaker |
| **CRC** | Cyclic Redundant Check |
| **CSMA/CD** | Carrier Sense Multiple Access with Collision Detection |
| **CT** | Current Transformer |
| **DA** | Distribution Automation |
| **DAC** | Centralised DAS, Distribution Automation Controller |
| **DANH** | Doubly Attached Node |
| **DAS** | Distribution Automation Systems |
| **De-capsulation** | A process of unwrapping or decoding the data after it has been transmitted. |
| **LAN** | Local Area Networks |
| **Destination Address** | Recipient Address |
| **EMI** | Electromagnetic Interference |
| **Encapsulation** | A process where data are wrapped with the necessary protocol information before it is transmitted |
| **FCS** | Frame Check Sequence |
| **FDDI** | Fiber Distributed Data Interface |
| **FSM** | Finite State Machine |
| **Gbps** | Gigabits per seconds |
| **GE** | General Electric |
| **GOOSE** | Generic Object Orientated Substation Event |
| **GPS** | Global position system |

University of Cape Town

| | |
|---|---|
| **GSSE** | Generic Status Substation Event |
| **HSR** | High availability Seamless Redundancy |
| **IEC61850 standard** | International standard called “communication networks and systems for power utility automation” |
| **IED** | Intelligent Electronic Device |
| **IF** | Interface |
| **IP** | Internet Protocol |
| **ISO** | International Standards Organisation |
| **LAN** | Local Area Network |
| **layer 1** | Physical layer |
| **layer 2** | Data Link layer |
| **layer 3** | Network layer of Open System Interconnection reference model |
| **layer 4** | Transport layer of Open System Interconnection reference model |
| **layer 5** | Session layer of Open System Interconnection reference model |
| **layer 6** | Presentation layer of Open System Interconnection reference model |
| **layer 7** | Application layer of Open System Interconnection reference model |
| **MAC Address** | Media Access Control address |
| **Mbps** | Mega Bits per second |
| **MMS** | Manufacturing Message Specification |
| **MU** | Merging Unit |
| **Multicasting** | Transmitting messages to a group nodes in a network, but not all nodes |
| **NCC** | Network Control Centre |
| **NIC** | Network interface Card |
| **NPS** | Negative Phase Sequence |
| **OSI reference model** | Open System Interconnection reference model |
| **Phase-I relays** | Electro-mechanical relays |
| **Phase–II relays** | Electronic relays |
| **Phase–III relays** | Numeric relays |
| **Phase–IV relays** | Micro-processor-based relays |
| **Peer-to-peer communication** | each layer of OSI reference model at the source side communicate with its peer at the destination side. |

University of Cape Town

| | |
|---|---|
| **PCM** | Protection, Control and Metering |
| **PDU** | Protocol Data Unit |
| **PRP** | Parallel Redundant Protocol |
| **PSE&G** | Public Service Electric and Gas Company |
| **RSTP** | Rapid Spanning Tree Protocol |
| **RTU** | Remote Terminal Unit |
| **SA** | Substation Automation |
| **SEF** | Sensitive Earth Fault |
| **SMV** | Sampled Values |
| **SNTP** | Simple Network Management Protocol |
| **Source Address** | Sender Address |
| **Tnom** | Nominal Time |
| **STP** | Spanning Tree Protocol |
| **STP** | Shielded Twisted Pair |
| **TCP** | Transmission Control Protocol |
| **TCP/IP** | Transmission Control Protocol/Internet Protocol |
| **UDP** | Datagram Protocol |
| **UDP/IP** | User Datagram Protocol / Internet Protocol |
| **Unicasting** | Transmitting messages to one node in a network |
| **UTP** | Unshielded Twisted Pair |
| **VT** | Voltage Transformer |

University of Cape Town

# 1. Introduction

## 1.1 Background

Substation Automation (SA) and Distribution Automation (DA) describe various methods of automatically and rapidly isolating faults, restoring power, monitoring supply and demand, and maintaining and restoring electricity substation or distributed resources for more reliable delivery of electric power. Currently, in many networks, the role of isolating a fault and restoring electricity supply is performed by operators, and the monitoring functions are performed by Supervisory Controls and Data Acquisition (SCADA) systems [1].

What makes SA and DA easy to implement is the use of intelligent electronic devices (IEDs), which are also called Phase–IV relays in the Eskom Distribution network. IEDs have the capability to perform protection, control and metering (PCM), and monitoring functions. With the use of IEDs, both SA and DA can monitor and control power delivery systems in electricity substations and on the distribution network re-closers (breakers on the distribution poles), thereby reducing the duration and number of power outages [1].

The IEDs are intelligent in the sense that they are capable of achieving more than an standard relay can, viz. protection functions, metering or measuring of analogous values. Other achievable IED functions are local control of breakers, such as close and open, auto reclose “on” and “off”, and sensitive earth fault (SEF) “on and off”.

IED protocols or languages include the IEC61850 protocol, which is used in SA. IEC61850 is an international standard which follows the Open System Interconnection (OSI) reference model with regards to communication. The messages used in IEC61850 include the Generic Object Orientated Substation Event (GOOSE), Generic Status Substation Event (GSSE) and the Sampled Raw Data, which are the sampled currents and voltages. IEC61850 follows the OSI model, the GOOSE, GSSE, and sampled values messages also follow the OSI reference model. GOOSE and the sampled raw data messages use only three layers of the OSI reference model; which makes these messages faster relative to other messages using the OSI reference model. The GOOSE and sampled raw data messages are therefore said to be faster than the conventional physical contacts. The conventional physical contacts can either be the contact outputs or contact inputs on the IEDs (Phase–IV relays), Phase–III, Phase-II or Phase-I relays. An example of the contact output is the trip contact to the circuit breaker.

The generations/phases of relays in accordance to Eskom Distribution is discussed to illustrate the evolution of relays up to IEDs. The first generation of protective relays was invented more than 100 years ago and has evolved to more sophisticated micro-processing-based relays [1:2-13]. The first generation of relays was electro-mechanical (phase-I). These relays had limitations, since each relay had a single functionality which was either over-current or earth fault. When bulk electricity is transmitted from one place to another, it is sometimes transmitted in three-phases, which are red, white and blue phases, or A, B, or C phases. The indications on these relays were confusing, as they were in the form of flags that were not easily isolated to specific phases. For instance, an

University of Cape Town

operator checking a fault incidence at a substation would not be certain on which phase the fault occurred, since sometimes all three phases (red, white and blue) of the relays would have operated but at different times and for different fault types. The phase-I generation of relays did not have any computing, memory or communications capabilities [1]. The second generation of relays was called electronic (phase-II) relays. This generation of relays was an improvement over phase-I relays. They consist of more than a single function, such as over-current and earth fault. The third generation of relays was the numeric (phase-III) relay, which are an improvement over phase–II relays. They also consist of over-current or earth fault functionality and additionally incorporate negative phase sequence (NPS), auto-reclose (ARC) and sensitive earth fault (SEF) functions. The latest generation of relays is called micro-processor-based (Phase-IV) relays, or IEDs.

Phase-IV relays are multifunctional. For instance, in the case of feeder protection, where the protection relays are used to protect a feeder or a line in a network, one relay can perform feeder protection functionalities, measurements of analogous values for metering purposes, local control of a breaker, auto-reclose (ARC) and SEF and monitoring of the power system equipment. The control and monitoring functions are required by Supervisory Controls and Data Acquisition (SCADA) departments to remotely determine network activity, for instance, when the breaker trips/opens following a fault condition. An example of monitoring breaker conditions is when breakers take more than the allowed set time (in most cases 100 milliseconds), indicating that the breaker is slow to clear the fault. In this instance, the phase-IV relay monitoring function should notify SCADA; hence enabling relevant persons to undertake necessary field maintenance.

Figures 1.1 and 1.2 show Phase–I and Phase–IV relays respectively. Phase-I relays in Figure 1.1 are electro–mechanical and have no form of memory and communication. This means that one cannot communicate with the relay to retrieve any information after incidents or fault conditions in order to perform an investigation. Phase–IV relays on the other hand have memory and communication capabilities and can perform protection, control, monitoring and metering functions.

ersity of Cape Town

**Figure 1.1 Phase–I relay with components that lack computing, memory, or communications capabilities required for substation automation [1]**

**Figure 1.2 Phase–IV relays also called IEDs are used for Substation Automation [1]**

University of Cape

The Phase–I and Phase–IV relays in Figure 1.1 and 1.2 use physical contacts, which are also used by the Phase–II and Phase–III Relays. Moreover the Phase–IV relays support the IEC61850 protocol, which is used in SA. The messages used in IEC61850 are the GOOSE and Sample Raw data messages.

The challenge is to prove or disprove whether the GOOSE messages are indeed transmitted faster than the operation time of physical contacts. If the GOOSE messages are found to be slower than the physical contacts, the factor that is influencing the performance of GOOSE messages needs to be determined. The influence of sampled values traffic on the transmission time of GOOSE messages also needs to be determined.

## 1.2 Research Objectives

The objectives of the research are to:

- Investigate what has been implemented in South Africa and the World with regard to Substation Automation, especially the performance of GOOSE messages.

- Perform lab experiments to:

  - Compare the speed of GOOSE messages with the speed of hardwired contacts of the relays.

  - Test a Breaker fail function using GOOSE messages instead of using the conventional hardwired contacts.

- Establish the interoperability of IEDs from different vendors using GOOSE message sharing.

## 1.3 Key questions

The key questions are stated as follows:

**Key question 1**

- With regard to protection performance, are GOOSE messages faster than the physical hardwired contacts?
  **Sub-question:** If not, what are the factors that can affect the performance of GOOSE messages?

**Key question 2**

- How does the introduction of sampled values traffic in the network affect the performance of GOOSE messages?

**Key question 3**

- Can GOOSE messages be used to replace the conventional protection such as breaker fail bus strip operation?

## 1.4 Methodology

The first task is to review literature on what has been implemented concerning Substation Automation (SA), especially the performance of GOOSE messages. This is used to guide the empirical research in the form of modelling configurations to test the research questions.

Secondly, to answer the first key questions, laboratory experiments were set up to compare the speed of the physical hardwired contacts and the GOOSE messages. These experiments were completed using two Generic Electric (GE) F35 series relays (relay-1 and relay-2), an Omicron test set, laptop and a Human Machine Interface. All these devices communicated via an Ethernet switch.

To answer the first key question, a common starting point was created using an Omicron test set. This is to ensure that when the experiment is performed everything that is compared has the same reference point. To create a common starting point, an Omicron Binary output contact was closed, resulting in closing the F35 relay-1 contact input. From the relay-1 contact input, the contact input was configured independently to the relay-1 physical contact and GOOSE outputs.

The F35 relay-1 physical contact output and GOOSE output was then be used as inputs to the F35 relay-2. Therefore, the F35 relay-1 physical contact output and GOOSE output were configured to the F35 relay-2 physical contact input and GOOSE input respectively. The inputs of relay-2 were then be configured to its contact output, the relay-2 physical contact input was configured to its physical contact output, and relay-2 GOOSE input is configured to its GOOSE output.

University of Cape Town

To simplify the comparison between the performance of physical contact signals and the GOOSE messages, only the GOOSE and the physical contact outputs results of relay-2 was analysed in detail; and not the results in between the Omicron Binary output contact and the relay-2 outputs. What happens in between was considered as the contributing factors to the performance of GOOSE messages and physical contact outputs signals of relay-2. This will help to answer the first key question.

To answer the second key question, a lab experiment was conducted where the sampled value traffic was included.

To answer the third key question, a lab experiment was conducted where a conventional breaker fail–bus strip operation is replaced by GOOSE messages, and where the relays are directly connected via an Ethernet fibre optic cable.

## 1.5 Scope and Limitations

This dissertation defines SA and the importance of IEC61850 standards. It further defines the topologies that can be used in SA and their importance. Existing developments are also discussed, and finally lab experiments are performed in order to answer the key questions.

This research is limited to the performance of GOOSE messages in comparison to the physical relay contacts; and the use of GOOSE messages to replace conventional protections such as the Breaker-fail operation. It does not include data security aspect of IEC61850.

## 1.6 Thesis outline

Based on the key questions, a review of the literature is carried out in Chapter 2. Chapter 3 explores network topologies and protocols that can be used in SA. Chapter 4 discusses what has been implemented in terms of SA. Chapter 5 presents the results of experiments and discussion. Conclusions and Recommendations are presented in Chapter 6.

University of Cape Town

# 2. Substation Automation

## 2.1 IEC61850 Standard

IEC61850 is an international standard, which is called “communication networks and systems for power utility automation”. It defines the communications between devices in the substation and the related system requirements, such as message performance (The speed of the trip command messages, and Sample Raw data messages, such as current and voltages) and information security in an automated network. It supports all substation automation functions and their related engineering [2:1-5].

Figure 2.1 illustrates a typical Substation Automation (SA) communication diagram.

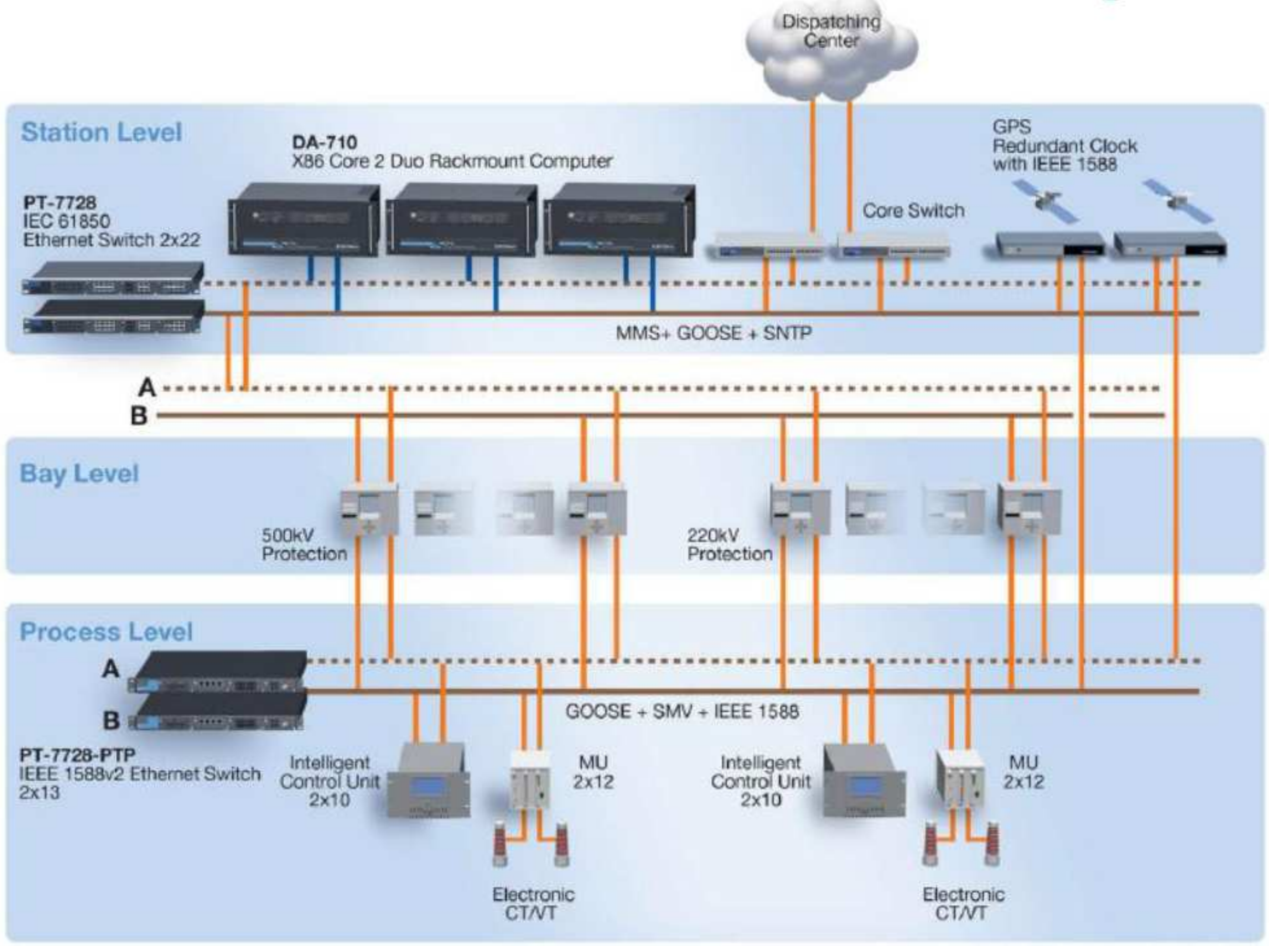


**Figure 2.1 SA communication diagram in a substation [3]**

In Figure 2.1, MMS means Manufacturing Message Specification, SNTP means Simple Network Management Protocol, SMVs means Sampled Values, CT means current transformer, and VT means Voltage Transformer.

From Figure 2.1 it can be seen that the SA communication has a station level, bay level, and process level.

**Station level incorporates the following:**

- Communication between intelligent electronic devices (IEDs) and master stations.
- Data polled by master stations from IEDs or randomly sent by IEDs.
- Intra-IED data exchange through multi-casting Generic Object Orientated Substation Event (GOOSE) messages.
- The messages that are transmitted on this level are Manufacturing Message Specification (MMS) and GOOSE messages; and one of the time synchronizing protocols that can be used is Simple Network Management Protocol (SNTP) [4].
- The global positioning system (GPS) device can also be used to provide synchronism.

**Bay level incorporates the following:**

- Inter communication of data between the process, bay and station levels.
- Intra-IED data exchange of the bay level messages [4].

**Process level incorporates the following:**

- Communication between plant equipment and IEDs via merging units (MUs).
- Exchange of sampled values (digitized measurements) via Merging Units or digitalized Transduces. The MU IED first processes and combines the signals from the electronic current transformer (CT) and Voltage Transformer (VT), then transmits the digital voltage and current output to the process.
- Control of data exchange between IEDs or Intelligent Controller Units and plant equipment messages.
- The messages that are transmitted on this level are GOOSE, Sampled Values (SMVs), which are sampled analogue values, such as currents and voltages, and IEEE 1588 precision time protocol is used to provide fault tolerant synchronization for different clocks along the same network. [4, 32].

## 2.2 Open system interconnection reference model of IEC61850

The IEC61850 standard follows the Open System Interconnection (OSI) reference model. The GOOSE, Generic Status Substation Event (GSSE) and the Sampled Raw data are some of the messages used in IEC61850. Since IEC61850 follows the OSI model, the GOOSE, GSSE and Sampled values messages follow the OSI as well (see Figure 2.2).

University of Cape Town

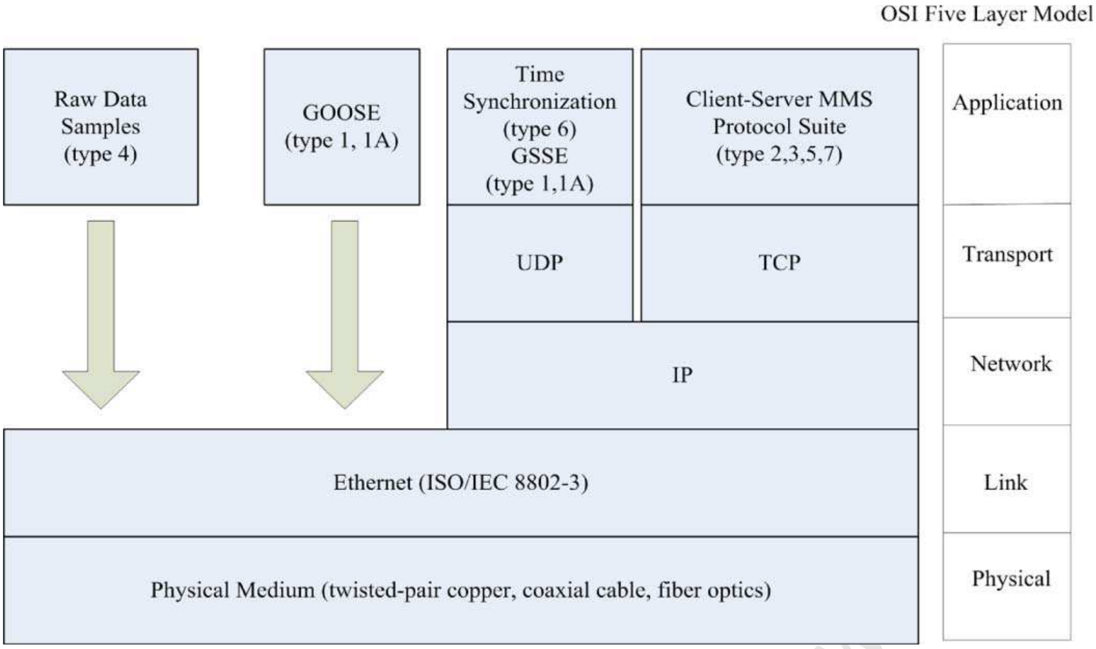


**Figure 2.2 IEC61850 communications stack with reference to the OSI model [5]**

OSI is a network architecture model that consists of seven layers, each of which has a specific function in data communication. The OSI model was developed by the International Standards Organisation (ISO) in order that complex network communications can be understandable to users. Although other models exist, most network vendors today relate their products to the OSI model [6].

In Figure 2.2, it can be observed that both the GOOSE and the Sampled Raw data messages use only three layers of the OSI reference model. These are the Physical layer (layer-1), Data Link layer (layer-2) and Application layer (layer-7) [6]. The fact that only three layers are used is the reason why the GOOSE and the Sampled Raw data transmission time is claimed to be shorter than all the other messages in the OSI communication stack. The data string of data such as GSSE is longer than that of GOOSE messages, since GSSE goes through the IP (Internet Protocol) and TCP (Transmission Control Protocol) layers, therefore its data string will have the recipient's and sender's IP addresses. Figure 2.3 illustrates a simple Ethernet frame or data string, the same as is used for GOOSE and Sample Raw data messages.

University of Cape Town

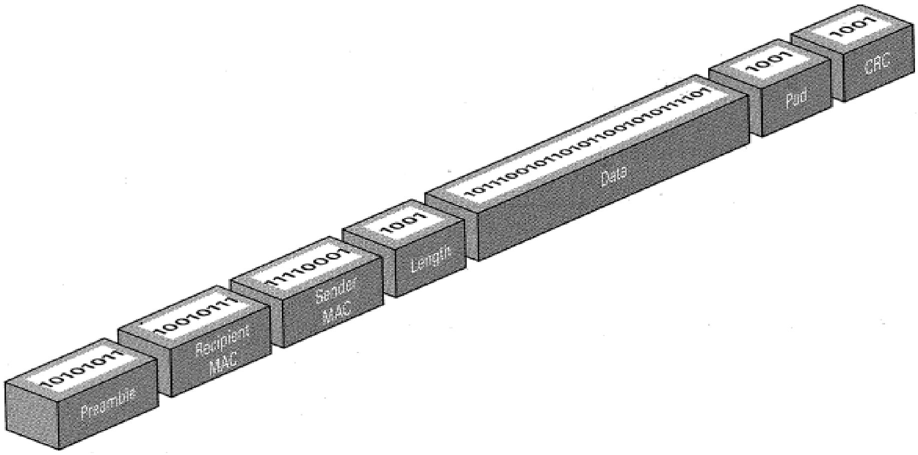


**Figure 2.3 A simple Ethernet frame [6].**

The following section defines and explains the functions of different fields of an IEEE 802 Ethernet frame. The reason for going into detail about the GOOSE and Sampled Raw data messages is because the lab experiments were based on GOOSE and Sampled Raw Data messages, so it is important that anything that might have a significant impact on the performance of these messages is explained or clarified. The different fields of an IEEE 802 Ethernet frame are:

- **Preamble**: Ethernet frames begin with a preamble, which is a 64 bit series of 1s and 0s. The preamble provides an indication to the Network interface Cards (NICs) that a frame is coming and where the exact message is starting [6:11-209].

- **Destination Address**: This field contains the Media Access Control (MAC) destination address, also called the physical address. The length of the MAC 6 octets or special 48 bit binary number, is normally expressed as a hex-decimal. The destination address can be unicast (single node), multicast (group of nodes) or broadcast (all nodes) [6:209].

- **Source Address**: This field contains the 6 octets MAC source address. The source address is supposed to be the only unicast address of the transmitting Ethernet station. However, due to an increase in the virtual protocols which are being introduced, there is the possibility of a single MAC source address being used by different entities to identify themselves [6:11-209].

- **TPID** (Tag Protocol Identifier): This file contains the 2 octets of tagging. This tagging on the frame is added to describe the priority of the message, e.g. high priority messages for GOOSE messages or Raw data messages.

- **Length:** The maximum length of an Ethernet frame is 1500 bytes of data. This includes the preamble, recipient's and sender's MAC addresses, the actual data that is transmitted, and CRC (cyclic redundant check) and the length itself. The minimum frame length is 64 bytes [6:11-209].

- **Data:** An Ethernet frame may carry up to 1500 bytes of data in a single frame, so the message or data should not exceed that size [6:11-209].

- **Pad:** This field is only inserted when it is necessary, as when the data frame minimum length is not met by the data. The minimum frame length must be 64 bytes [6:11-209].

- **Cyclic Redundant Check (CRC)/ Frame Check Sequence (FCS):** This sequence contains a 4 octets value that is created by the sending device and is recalculated by the receiving device to check for any errors in the frames [6:11-209].

## 2.3 Explanation of different communication layers of the OSI reference model.

**layer 1—Physical layer:** The Physical layer specifies the physical data rates, maximum transmission distances, physical connector types, and other similar attributes. An example

University of Cape Town

of the Physical layer is 100BaseFX, which specifies a fiber optic cable that has a maximum data transmission of 100 Mbps [6:11-209].

**layer 2—Data Link layer:** The Data Link layer provides reliable transportation of data across a physical link. The layer is concerned with the physical or MAC address. The MAC addresses are found on the NICs of each device which is connected to a network. This layer deals with network access, error notification, ordered delivery of frames, and flow control. Examples of layer 2 protocols include Ethernet and Token Ring. Ethernet switches operate on this layer [6].

**Ethernet:** Ethernet is the most widely used Local Area Network (LAN) technology. It defines the wiring and signalling standards for the Physical layer of the OSI model. Ethernet was originally standardized as IEEE 802.3 with a data transmission rate of 10 Mbps, which was the 10baseT technology. The 10baseT was very popular and it dominated the token Ring technology (defined below). Over time, newer versions of Ethernet were introduced to offer higher data rates. Fast Ethernet and Gigabit Ethernet support data rates of 100 Mbps and 10 Gbps (10000 Mbps) respectively, the fastest being the 10 Gbps [6]. LAN will be explained in depth later on in the report.

Some of the cables utilised with Ethernet are coaxial, twisted and fiber optic. An example of coaxial cable is 10Base2, where the 10 represents 10 Mbps, base represents baseband signalling and 2 represents the max distance of communication, which in this case is 200 metres. Coaxial is no longer used. Twisted cable comes in two forms, which are unshielded twisted pair (UTP) and shielded twisted pair (STP). UTP is less expensive and more commonly used, and STP is more expensive because of the braided metal shield used to protect the cable against electromagnetic interference (EMI), e.g. 10baseT or 100baseT. The maximum distance of twisted cables is 100 meters. Fiber optic cable can be used from 2 to about 40 kilometres. They are not susceptible to EMI and they have gigabits as a transmission rate [6].

Ethernet devices, e.g. Ethernet switches, compete for access to a network using a protocol called Carrier Sense Multiple Access with Collision Detection (CSMA/CD). Each device in the network senses or detects to check if there is traffic or data flow. If there is data flow already in the network, the device that is intending to send data will wait for some milliseconds to try again. These are done to avoid collision in the network [6].

**Token Ring:** Token Ring is another LAN technology. It consists of a token that is passed around a network. The token in essence goes through a circle in one direction. This is done to control the number of computers sending data at the same time to avoid collision, just as with CSMA/CD in Ethernet. Whichever computer wants to transmit looks for the token. Once it gets it, it can transmit data into a network. It first receives the token then discards it and replaces it with its own data frame. When another station receives data addressed to it, it will mark the frame as received and pass it back out onto the network. When the frame is received back by the originating station it will be discarded and a new token will be released to the network. Token Ring devices are often more costly than Ethernet devices. There are many tools, utilities and suppliers of devices for Ethernet, while the choices for Token Ring are much more limited [6, 17].

University of Cape Town

**layer 3—Network layer:** The Network layer provides connectivity and path selection (routing functionality) between two devices' systems that might be located on geographically separated networks. Routers and layer 3 switches operate at this layer. The Network layer is concerned with logical addressing. It performs fragmentation and defragmentation and reports delivery errors. An example of layer 3 protocols is Internet Protocol (IP) [6, 20].

**layer 4—Transport layer:** The Transport layer provides transparent transmission of data between the IEDs. It segments data from the sending device system and reassembles it into a data stream on the receiving device system [6]. The Transport layer is concerned with integrity of data, quality of service and reliability of transport between two devices. Examples of layer 4 protocols are the Transmission Control Protocol (TCP), and segments and User Datagram Protocol (UDP) [6].

**layer 5—Session layer:** The Session layer establishes, manages and terminates sessions between two communicating devices. This layer provides its services to the Presentation layer. It also synchronizes or controls dialogue or connection between the two hosts' Presentation layers, and manages their data exchange, ensuring a proper inter-host communication. It carries out reporting for Session layer, Presentation layer, and Application layer problems [6, 20].

**layer 6—Presentation layer:** The Presentation layer ensures that the information that the device's Application layer sends out can be read by the Application layer of another device. One of the most important tasks of this layer is encryption (conversion of data into a form that cannot be easily understood by unauthorized users) and decryption (changes to data so that are meaningful to all users) [6, 11].

**layer 7—Application layer:** The Application layer is the layer that is closest to the user. It provides network services to the user's applications. It differs from the other layers in that it does not provide services to any other OSI layer; instead it provides services only to applications outside the OSI model. It selects the appropriate services to be applied for the end user. Examples of Application layer services are spread sheet and word-processing programs [6].

The application, presentation and Session layers are concerned with application issues; whereas the lowest four layers are concerned with data-transport issues.

## 2.4 Advantages of using the OSI reference model

The advantages of using the OSI reference model are as follows [6]:

- The OSI reference model standardizes network components to allow multiple-vendor development and support.

- It allows different types of network hardware and software to communicate with each other, which is called interoperability. Interoperability is the communication between IEDs from different vendors.

University of Cape Town

- It prevents changes in one layer of communication from affecting the other layers so that they can be developed more quickly.

- It breaks network communication into smaller components to make learning easier.

Since the IEC61850 communication standard follows the OSI model, which addresses interoperability, it automatically addresses the issue of interoperability. This means that one does not require only the IEDs from the same supplier or vendor to be able to implement substation automation. One example of interoperability is peer-to-peer communication of IEDs from different vendors [2]. Peer-to-peer communication means each layer of the OSI reference model at the source side must communicate with its peer at the destination side [6].

During peer-to-peer communication, the protocols at each OSI layer exchange information between peer layers. Data that is sent from one IED to another IED are called protocol data units (PDU). Figure 2.4 illustrates peer-to-peer communication.

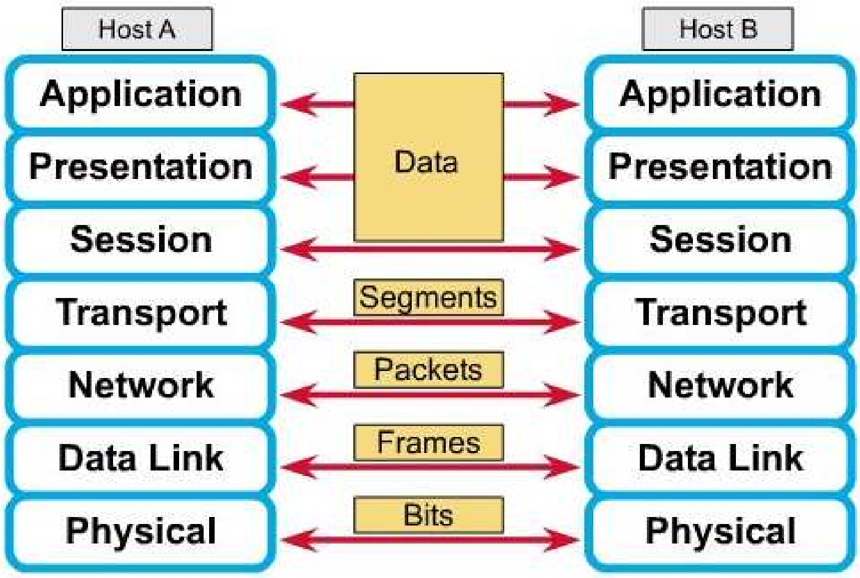


**Figure 2.4 Peer-to-peer communication [7]**

In peer-to-peer communication, data packets on a network originate at a source and then travel to a destination. The data names evolve with the layer they are occupying, e.g. at the physical layer, data are called bits; at the data link layer, data are called frame; at the Network layer, packets, and so on (see Figure 2.4).

Each layer depends on the service functions of the OSI layer below or above it, e.g. the Data Link layer depends on the Physical layer, and the Network layer depends on the Data Link layer and vice versa. To provide the services, the lower layer uses encapsulation to put the PDU of the upper layer into its data field. Encapsulation is a process where data are wrapped with the necessary protocol information before they are transmitted. Each layer adds a header or a trailer applicable to it so that the frame is ready for the next layer (refer to Figure 2.5) [6].

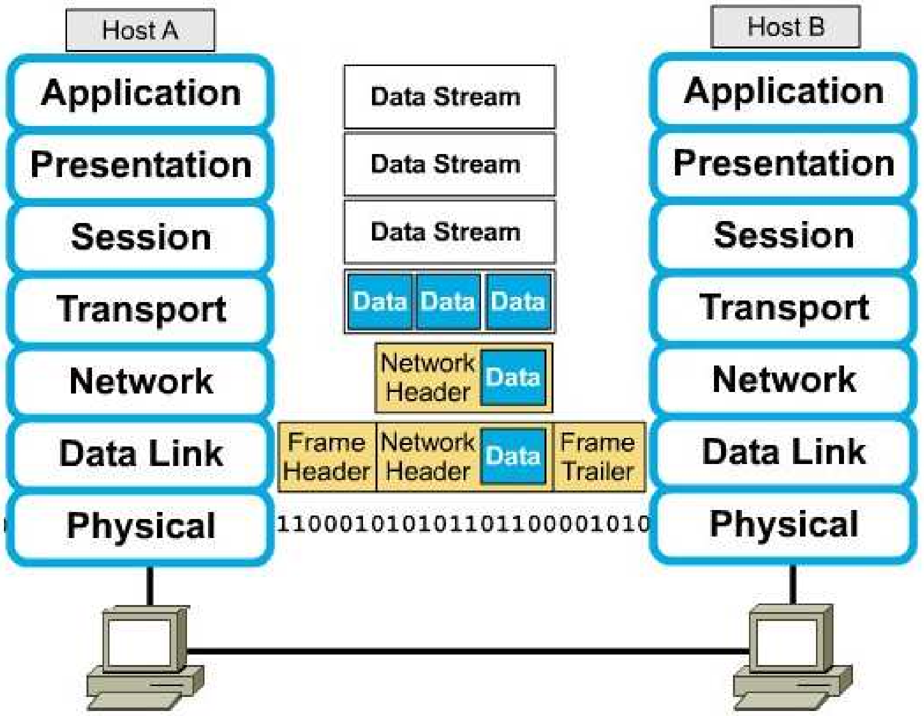


**Figure 2.5 Encapsulation Process [7]**

University of Cape

## 2.5 Encapsulation and De-capsulation Processes

Encapsulation is a process where data are wrapped with the necessary protocol information before it is transmitted. It changes one type of network data to other data types. It occurs when a protocol that is on a lower layer receives data from a protocol at a higher layer, and the data are wrapped into a data format that is understood by the receiving protocol [8].

De-capsulation is simply the inverse of the encapsulation process. It is a process of unwrapping the data. The encapsulation process starts from the uppermost layer -the Application layer, to the lowest layer - the Physical Layer, while the de-capsulation process starts from Physical layer to the Application layer [8].

- **Encapsulation Process**

All communications on a network originate at a source and are sent to a destination. In Figure 2.5, the Host A is referred to as a source and Host B as a destination. If Host A wants to send data to Host B, data must first be encapsulated [6:11-209]. Therefore, as the data are passed down through the layers of the OSI model from the Application layer to the Physical layer, each OSI layer adds a header and also a trailer where applicable before passing it down to a lower layer. Data starts from an Application layer and is then taken to the Presentation layer where it is encrypted, then down to the Session layer, which controls dialogue between the source and the destination. From the Session layer data will be passed through to the Transport layer. When it reaches the Transport layer, it will be portioned into what is called **segments**. Data will then be taken to a Network layer where it will be attached with a header. The header contains information required to complete the transfer, such as the source (sender's) and destination (recipient's) logical addresses, which are the IP addresses. From the Network layer, the headers and trailers are placed on the **data packets** and then passed to the Data Link layer [6].

The header consists of the source and the destination MAC addresses. The trailer consists of the Frame Check Sequence (FCS), which is used to ensure proper delivery of the data so that the destination device/IED can determine if data that is sent is corrupted or not. A header is similar to the address on an envelope. An address is required on an envelope so that the letter inside can be delivered to the desired destination, so it is that any data transmitted in a network needs to have some kind of sender's/source and recipient's/destination address. Data on the Data Link layer are called **data Frames** [6]. The frame/data frame is then passed to the Physical layer. The Physical layer provides a service to the Data Link layer. The Physical layer encodes the Data Link frames into a pattern of 1s and 0s (**bits**) for transmission on the medium, e.g. Ethernet cable [6:11-209].

- **De-capsulation**

When the remote (destination) device receives a series of bits, the Physical layer at the destination device passes the bits to the Data Link layer for manipulation. On the Data Link layer, data are now in the form of data frames. The Data Link layer then performs the following functions [6]:

- Verifies that the MAC destination address on the data matches the device's MAC address or else the address must be an Ethernet broadcast address which is sent to all the devices connected to the source device. If neither of these situations is true, the data frame is discarded.

- If the data has an error, it will be discarded, and the Data Link layer will ask for the data to be re-transmitted. If the data do not have error, the Data Link layer will read and interprets the information in the Data Link header.

- The Data Link layer will then strip off the Data Link header and trailer and then pass the remaining data up to the Network layer based on the control information in the Data Link header.

Data will then be taken up the OSI model until the Application layers. The difference between de-encapsulation and encapsulation is that, with de-encapsulation, the headers and the trailers which were put on during the encapsulation process are taken off. The de-encapsulation process is similar to receiving a letter and reading the address on the letter to see if the letter is yours, and then removing the letter from the envelope if the address on the letter is yours.

The process of encapsulation and de-encapsulation takes time since the headers are wrapped on to the data and later taken off. With regards to GOOSE and Sampled Raw data messages and the Raw Samples values, the transmission time from the source to the destination device is faster – as the data do not go through all the layers of the OSI model. Data only go through the layers 1, 2 and 3 of the OSI model, which means less wrapping and unwrapping of headers.

University of Cape Town

# 3. Network Topologies and Protocols: Application to Substation and Distribution Automations

## 3.1 Network Topologies

Network topology is the arrangement of the various elements within a network, such as links and nodes of IEDs. They can be represented physically or logically. The physical topology refers to the layout of a network based on physical components, such as devices, cables and connectors, while the logical topology indicates how data flows within a network, regardless of its physical design [9].

Local Area Networks (LANs) are one form of wired network topology. They consist of computers, network interface cards, peripheral devices, networking media, and network devices. Their function is to make the local sharing of information between the devices possible [6:11-209]. A LAN connects devices that are relatively close in proximity, such as devices that are in the same building [5].

In Substation Automation (SA) the network topologies are important, as one needs to know the kind of topology and the benefit of using a particular topology. The network topologies will be discussed in detail in the subsections 3.1.1 to 3.1.5.

### 3.1.1 Star Network Topologies

A Star topology is designed with each device connected directly to a central network device, such as an Ethernet switch or a hub (see Figure 3.1). Data on a Star network passes through the switch before it travels to the destination device. A Star network requires more cables, as each device is connected independently to the central device. Star topology is one of the network topologies that are fault tolerant, since a failure in any Star network cable will only affect the device connected to the faulted cable and not the entire LAN. However, if the switch or the hub fails, the entire network fails [11].

University of Cape Town

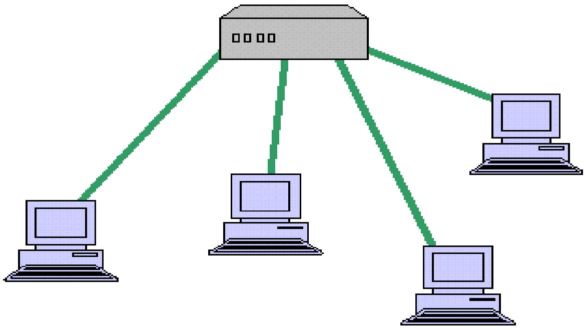


**Figure 3.1 Star Topology [12]**

### 3.1.2 Bus Network Topology

A bus network uses a single cable. This single cable provides a shared communication medium that interconnects all the devices connected to it (see Figure 3.2). Any device that requires to communicate with any other device on the network sends a broadcast message (i.e., a message addressed to all devices connected to the bus), but only the intended recipient will respond and process the sent data. The drawback of a bus network is that if a part of the cable is broken the whole network fails [11].

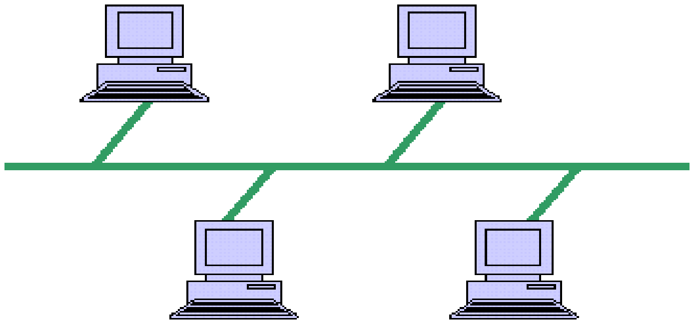


**Figure 3.2 bus Topology [13]**

### 3.1.3 Meshed Network Topology

A Meshed topology brings together the features of a bus, a Ring and Star topology. In a Meshed network there are some devices which are connected to two or more other devices in the network. Data that are sent can take any of several possible paths from a source to destination device. A Mesh network in which every device connects to every other is called a full Meshed network. A Meshed network where some of the devices connect to at least two other devices is called a partial Meshed network, as shown in Figure 3.3 [11]. The drawbacks of the Meshed topology are that the signal can end up trapped in a loop, not getting to the destination device, and a Meshed network can be chaotic when the network gets bigger.

University of Cape

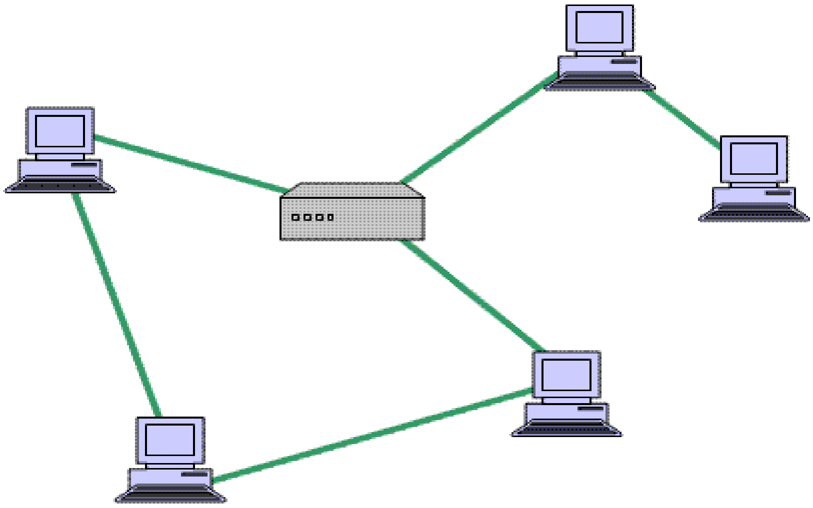


**Figure 3.3 Partial Meshed Network Topology [14].**

### 3.1.4 Tree Network Topology

Tree topology is an integration of multiple Star topologies together onto a bus topology (see Figure 3.4). In its simplest form, only hub devices are connected directly to the Tree bus, and each hub functions as the "root" of a Tree of devices. The Tree like structure allows you to have many servers on a network and a network can branch out into many other small networks. Similar to the Star topology, the Tree topology network is completely dependent on the device, a switch or a hub, which is on the main backbone of a network. If this device fails the entire network fails [11, 16].

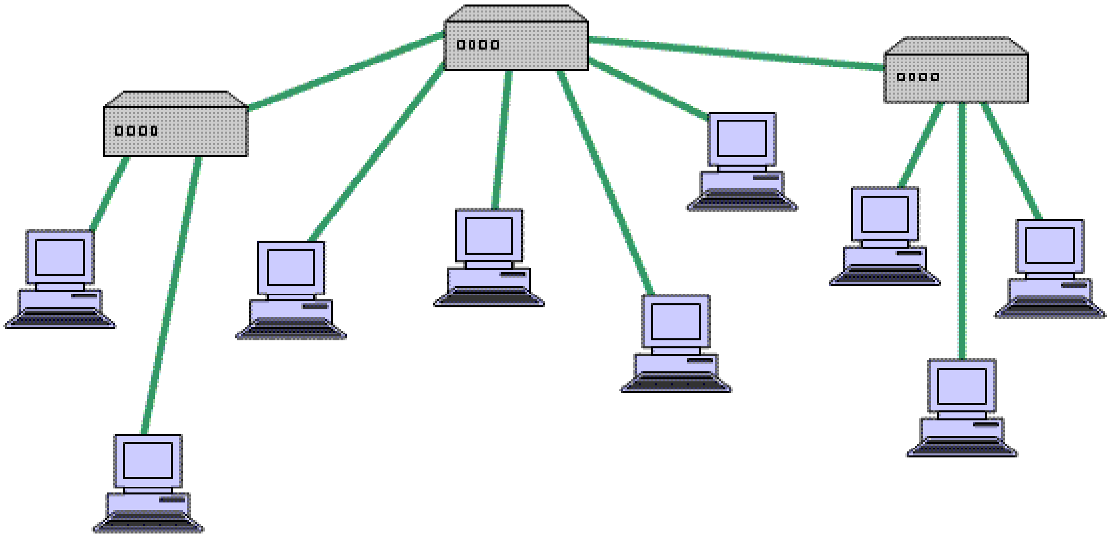


**Figure 3.4 Tree Topology [15]**

University

### 3.1.5 A Ring Network Topology

A Ring topology contains wiring that allows information to pass from one device to another in a circle or Ring fashion (see Figure 3.5). An example is a Fiber Distributed Data Interface (FDDI) network, where all the hubs are connected to each other in a Ring or a token Ring network. The drawback of the Ring Topology is that it is slow, since signals go through the circle in serial order, and if a cable goes faulty the whole flow is disturbed [11].

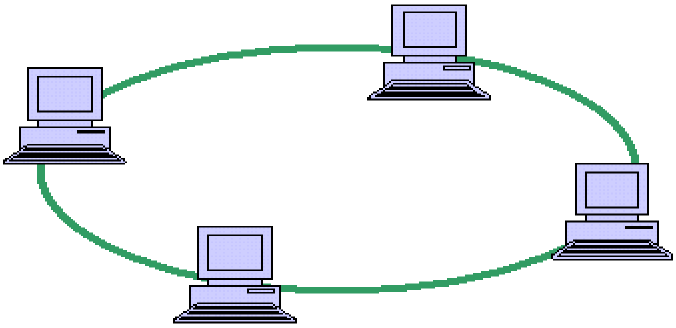


**Figure 3.5 Ring Topology [17]**

Table 3.1 illustrates the comparison of the Network topologies with regard to reliability and speed.

**Table 3.1 Comparison of Network Topologies**

| Type of Network Topology | Reliability | Speed |
|---|---|---|
| Star Topology | With regard to the cable connection, a Star topology is more reliable compared to a bus or a Ring network topology. This is due to the fact that each device is connected directly to a central network device such as an Ethernet switch.<br>If a single connection breaks or fails, only a single device will be affected and the rest of the devices will still be connected.<br>With regard to the central device such as a switch, if this device fails all the connected devices will lose connectivity. | A Star topology is fast when it comes to transmitting data from one device to another because data move from a device to another device through a central device. |
| bus Topology | Bus topology is not as reliable as the Mesh and Star topology. The reason being that it uses a single cable, which is shared by all devices connected to it. If the cable breaks all devices will be disconnected. | Similar to the Star topology, the bus topology is fast when it comes to transmitting data from one device to another. Data move from the source device into the cable and to the destination device. |
| Meshed Topology | Meshed topology is reliable as, in a network there are some devices which are connected to two or more of any other devices.<br>Data that are sent can take any of several possible paths from the source to the destination device. | Meshed topology is reliable .In this network one device can be connected to two or more devices.<br>Data that are sent can several possible paths from source to destination device. |
| Tree Topology | A Tree topology is similar to the Star topology. If a single connection fails, only a single device will be affected and the rest of the devices will still be connected.<br>With regard to the central device such as a switch, if this device fails all the connected devices will lose connectivity. | A Tree topology is relatively slower compared to the Star topology because of the fact that many devices are connected to one central device. However, it is faster than the Ring topology. |

University of Cape Town

| Type of Network Topology | Reliability | Speed |
|---|---|---|
| | | |
| Ring Topology | Ring topology is similar to the bus topology in the fact that all the devices are connected to a single cable. In a Ring network this cable is in a circular form though. | Compared to any other topologies, a Ring topology is very slow because when data are sent, it is first checked by the nearest device to the sending device. If the nearest device is not a receiver of the data, the data will be circulated by all other devices until they reach the receiver device. All the other devices also read the data to check if these particular data are sent to them. Again in a Ring topology, data go in one direction. Data do not check and used the shortest route [17]. |

## 3.2 Network Protocols

Network Protocols serves to improve network performance or an overall transmission time of data from the source to the destination.

### 3.2.1 Spanning Tree Protocol

The Spanning Tree protocol (STP) is a networking standard, as defined by the IEEE 802.1d or IEC/ISO 8802.1d standard. The purpose of Spanning Tree is to prevent looping in the LAN and to select the fastest network links if there are redundant links in a network. In the event that a link in a network fails, STP will choose an alternate link, provided that an alternative link exists. This is how a network can recover itself. Spanning Tree is a layer 2/Data Link layer protocol [18].

Spanning Tree works by first using an algorithm to find redundant links in the LAN and selecting the best paths. The original goal of STP is to put all links in either a forwarding or blocking state. The links without a redundant link and the best links of those which have redundant links would be in a forwarding state. The redundant links that are not as good as the selected links would be in a blocking state [18].

Spanning Tree cannot use multiple links to the same destination. Therefore, there is no load-sharing feature with Spanning Tree. Any redundant link that is not the best alternative link will be blocked or made inactive until the primary link fails. When the primary link fails the alternative link that is inactive at the time of failure will be activated. This is done to simplify the networking reconfiguration - to make it loop-less. One needs to bear in mind that a network does have some delays when it is reconfiguring and this can have a slight effect on the total performance of the STP.

The more complex and advanced STP is the Rapid Spanning Tree Protocol (RSTP). Similar to the STP, RSTP is very flexible and can convert a Meshed network protocol into a simple logical Tree, but does so faster [18]. An RSTP works in such a way that data are

University of Cape Town

transmitted from one end-node through the bridge or the bridges, depending on the shortest route, to the other end–node (see Figure 3.6).

Figure 3.6 illustrates the Rapid Spanning Tree Protocol (RSTP)

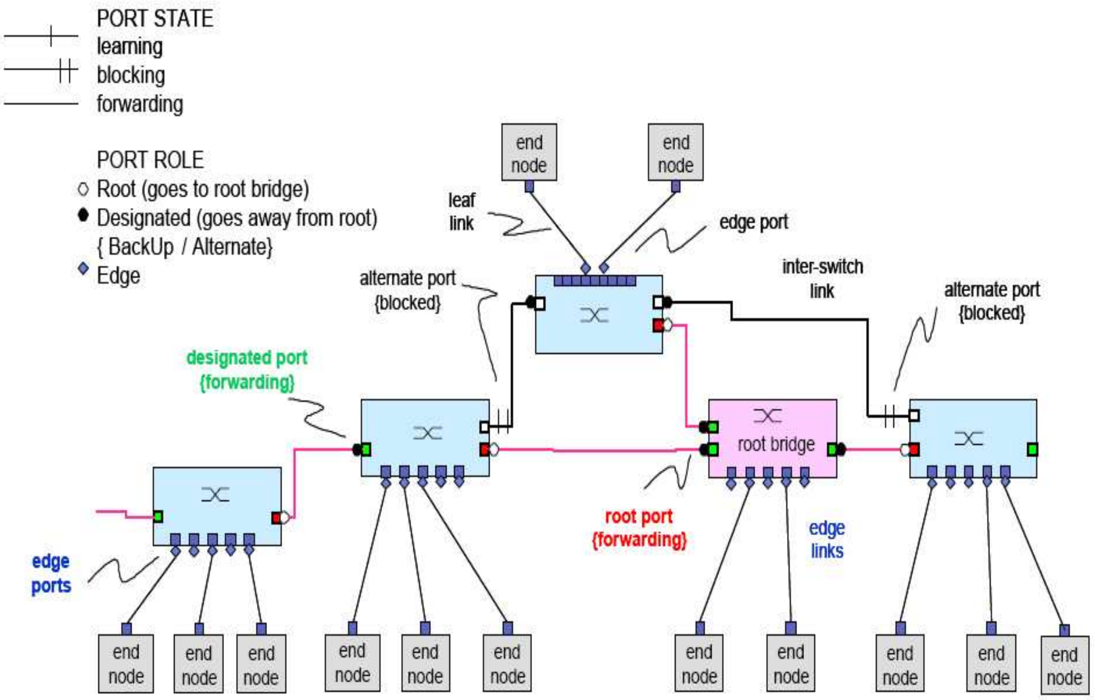


**Figure 3.6 Rapid Spanning Tree Protocol [19]**

University of

The criteria used to decide if a link should be in a forwarding state is as follows [19]:

- A root bridge is chosen. The root bridge is selected based on bridge identity (usually the MAC address) and a priority. By default, all priorities are the same and the switch with the lowest MAC address will become the root bridge.
- All the links connected to the root bridge are put in a forwarding state; the links in pink colour are all in a forwarding state (see Figure 3.6). The links in black are not in a forwarding state.
- For other bridges that are not connected to the root bridge, the port that is closest to the root bridge is put in a forwarding state.
- The port that is preferred but not connected to the root bridge is called a designated port, while the port that is connected to the root bridge is called a root port.
- The ports that are on the last connected bridge in a network are called an edge port.
- The Ethernet interface on the designated bridge is called the designated port.
- The end nodes represent IEDs that are connected to the network.

### 3.2.2 Parallel Redundant Protocol

The Parallel Redundant Protocol (PRP) is specified in the IEC62439 – 3 standard clause 4 as a protocol that offers a continuous failover. It follows a different approach compared to the reconfiguration protocols like the STP or RSTP. It has two independent routes for transmitting data [14, 18]. The frames are replicated by the sending node and transmitted over two independent routes of a network (see Figure 3.7). The two identical messages will then arrive at the receiving node and the message that arrives first will be executed while the one that arrives later will be discarded. In PRP there is no distinction between the relay-1 and the relay-2 path. To achieve the replication, PRP consists of two virtual nodes and these nodes facilitate the transmission of data in two independent routes of a network. The double nodes in PRP are called a doubly attached node (DANP). The P in DANP represents PRP [14, 18]. The advantage of this PRP is that it is faster than the STP or even RSTP in that it does not waste time during the reconfiguration time, but it is more costly than the STP.

Figure 3.7 illustrates the PRP topology. SAN in the figure stands for Singly Attached Node

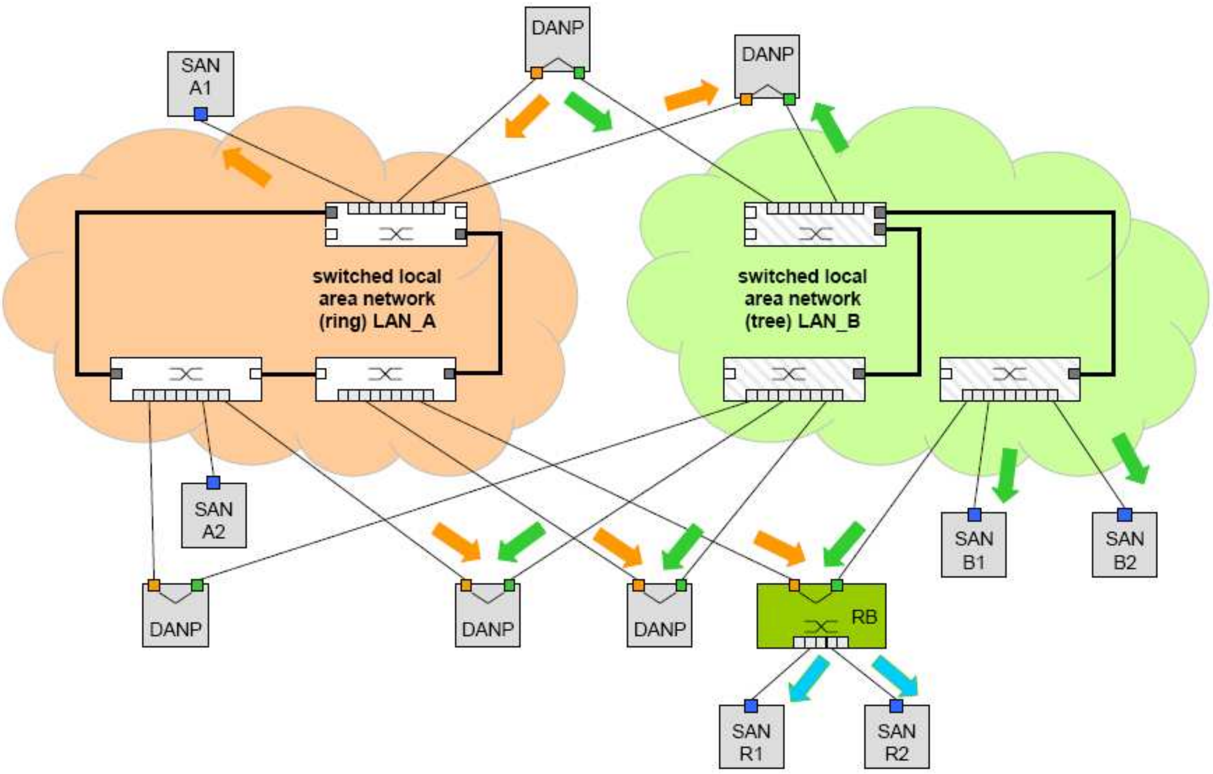


**Figure 3.7 Parallel Redundancy Protocol [19]**

### 3.2.3 High availability Seamless Redundancy

The High availability Seamless Redundancy (HSR) is specified in the IEC62439 – 3 standard clause 5 as a protocol that offers a continuous failover [28]. The HRS used the principle of PRP but it achieves redundancy through only a single additional link, meaning that the HSR is roughly half the infrastructure of the PRP. Each node in the HSR has a minimum of two ports and the nodes are daisy chained (connected in a ring). Similar to the PRP, the HSR uses the doubly attached nodes called DANH - H at the end represents HSR. Each node must be able to forward frames from a port at the wire speed, and this requires a bridge at each node, see Figure 3.8 [14, 18]. Similar to PRP, the frames in HSR are replicated by the sending node and transmitted over two independent routes of a network, which is clockwise or anti- clockwise in this case. The two identical messages will then arrive at the receiving node and the message that arrives first will be executed while the one that arrives later will be discarded. Figure 3.8 illustrates the High availability Seamless Redundancy Principle.

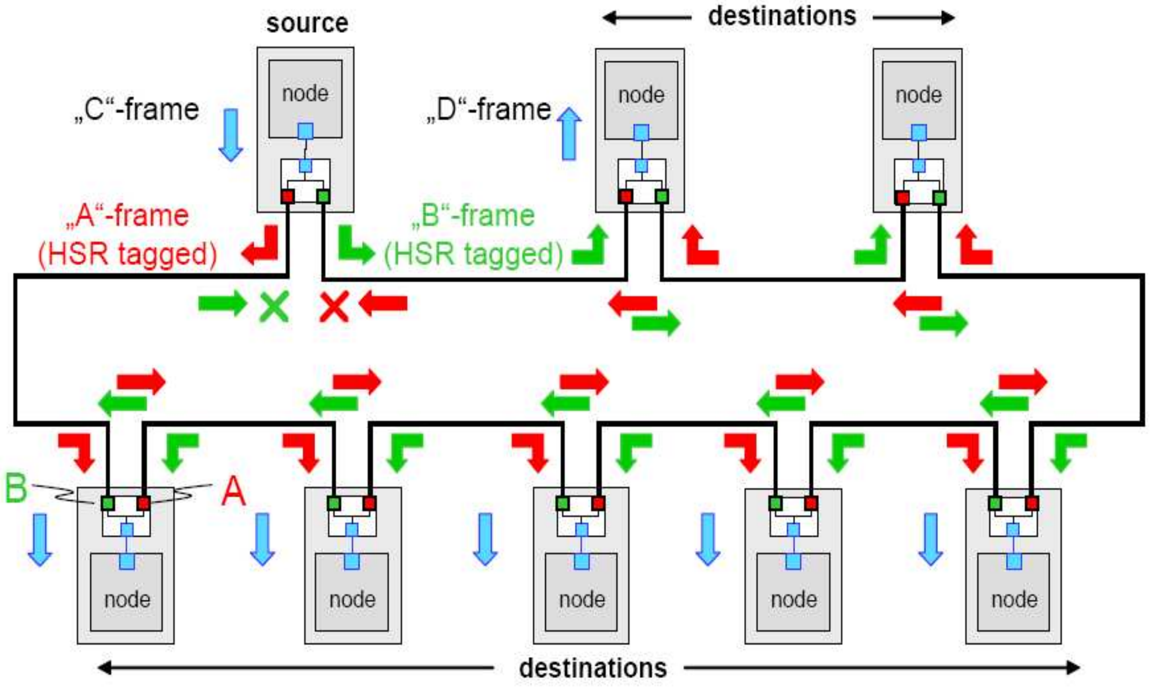


**Figure 3.8 High availability Seamless Redundancy Principle [19]**

Univers

## 3.3. Examples of Substation Automation or Distribution Automation Implementation

Distribution Automation (DA) describes various methods to automatically and rapidly isolate faults, restore power at the same time as monitoring supply and demand, to ensure reliable delivery of power supply to customers. Even though there are many IEDs installed on power systems to date, many have not been used for DA. However, there are a number of entities which took initiative to ensure that the power supply that they provide for their customers is reliable and of good quality, which they accomplished using IEDs [1].

DA not only reacts to maintain or restore stability but also assesses all available alternatives to use the optimal choice of automation. An example is evaluating pre-event demand and supply at all points on the system, and using this information and knowledge to predict short term demand profile changes. This lessens the incidents that can lead to voltage sags, spikes and interruptions [1].

IEDs used in electric power systems in utilities and industrial applications are multifunction devices, in addition to being PCM devices they are information and automation sources. These IEDs acquire power system data and then perform calculations to create a database with knowledge about the power system assets, such as transformers and breakers associated with them. Therefore, in addition to present power system values, these IEDs record information about the health, performance, and history of the overall power system and of the primary plant assets. Examples include monthly relay and breaker operation reports, meter reading reports, breaker condition reports, and transformer thermal monitoring reports [1].

### 3.3.1 Categories of Distribution Automation

There are two categories of Distribution Automation Systems (DASs) and these differ based on the available communications. One DAS design has a centralized decision engine and is called the Distribution Automation Controller (DAC). The other design is a distributed DAS, which has no controller but rather the communications and logic operate peer-to-peer among the IEDs. The DAC coordinates communications of data or messages between the IEDs [1].

Figure 3.9 and Figure 3.10 illustrate Centralized DAS and Distributed DAS respectively.

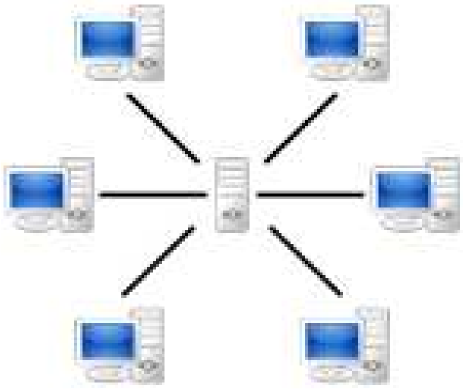


**Figure 3.9 Centralized DAS, server-based service model (not peer-to-peer), Using a Star topology [20]**

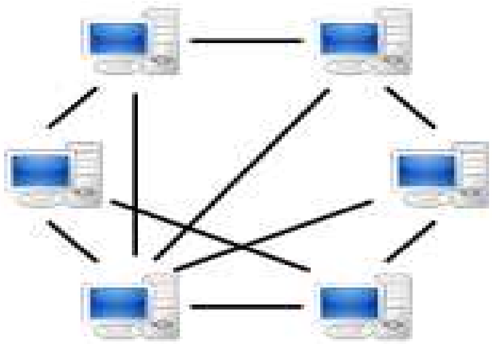


**Figure 3.10 Distributed DAS (peer to peer communication).Using a Meshed topology [21]**

In Peer-to-Peer communication, data packets on a network originate at a source and then travel to a destination. Both devices can be servers (initiate communication) or clients (respond to the initiated communication) at one point during communication. All the devices have the same capabilities and either party can initiate a communication session [1].

Below are the case studies that describe three in-service smart grid projects that are using specialized power system awareness devices in distribution applications [1].

- The first project consists of the centralized DAS techniques and it is a case study of American Electric Power (AEP), one of the largest electric power utilities in the United States.

- The second project consists of the distributed DAS (peer-to-peer) techniques and it is a case study of the Public Service Electric and Gas Company (PSE&G). The project is focused on innovative restoration and maintaining a stable system. This design has evolved around a standard loop-circuit configuration that is easily duplicated throughout large portions of their system.

### 3.3.2 Smart IEDs and Centralised DAS for American Electric Power–AEP

American Electric Power (AEP) uses a smart distribution automation controller (DAC), which is a centralised Distribution Automation System (DAS)**.** It is used to automatically react to faults and reconfigure a network via IEDs in substations and reclosers on distribution feeders. The system analyses and detects fault conditions, isolates the affected feeder section, and restores power to unaffected sections to effectively reduce outage times. The power delivery control system intelligently minimizes outage duration, and the number of affected customers, and then reports the actions taken [1].

The centralized logic performs analysis to detect permanent faults, broken jumpers and loss of substation source, and once it has detected the fault, it then isolates the faulted part of the network. The DAC evaluates system conditions to determine if any un-faulted zones are de-energized. If so, it automatically restores un-faulted zones using alternative sources of supply provided that the alternative source is available. Careful supervision is done before an alternative source of supply is used. Conditions such as network configuration, auto-reclose condition, and availability of supervisory control are monitored before an alternative source can be used. Alternate sources are selected based on zone load and available feeder capacity to avoid faults that occur due to the original fault. The DAC also changes settings groups within the IEDs so as to ensure that coordination between the protective devices is in accordance to the network configuration at that specific time [1:2-13].

Once utility personnel repair the affected zone, SCADA operators can issue a single command to systematically return the feeders to normal. The device settings are also automatically returned by selecting the corresponding setting group [1].

University of Cape Town

### Centralized Communication with Centralized Decision Making

The communication for centralized systems can be deployed in either a Star or loop topology, where the DAC acts as the controller for both the DA and the communications network (see Figure 3.11).

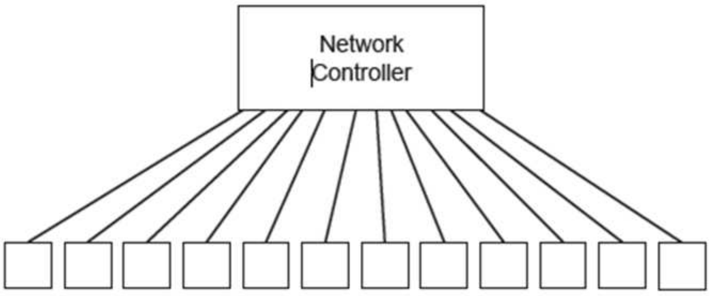


**Figure 3.11 Star Topology with centralized data collection and decision making through DAC [1]**

A Star configuration allows a network controller to use the appropriate protocol and data rate required to communicate with each IED. Each link can be a different collection of integrated protocols flowing at different speeds; they can even be over different transmission methods (i.e., radio, copper, or fibre serial cables) of communication [1]. The boxes in Figure 3.11 represent connected devices such as relays.

Figure 3.12 demonstrates the AEP centralized DAS that uses Meshed radios and multi-drop communication to collect information from all of the relays and breakers, in addition to making centralized automation decisions.

of Cape Town

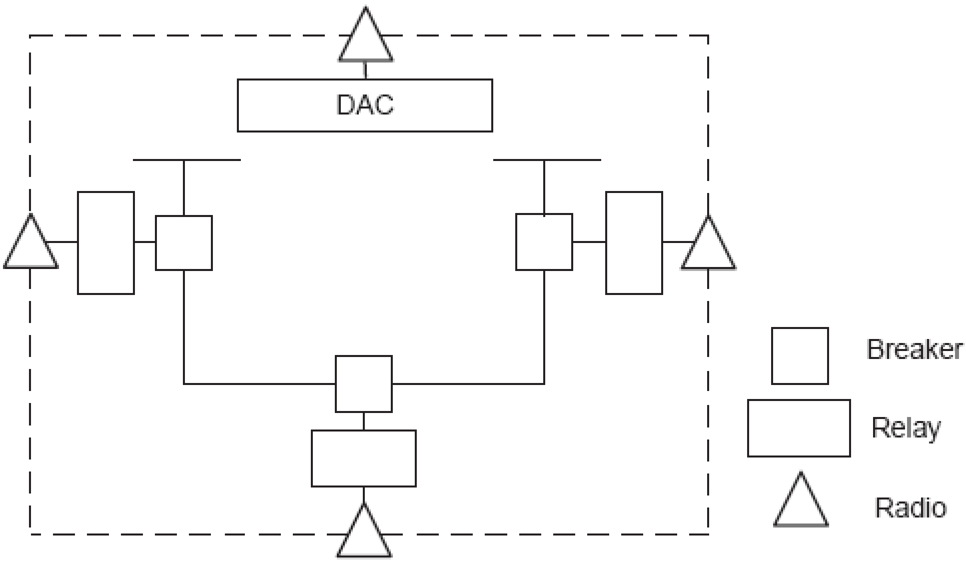


**Figure 3.12 Loop topology using centralized data collection [1]**

In Figure 3.12 data in the loop topology are transmitted through a multi-drop communication, from the breakers, relays through the radios to the DAC and vice versa. The decisions are made by DAC and not by individual relays. The square boxes represent the breakers, the rectangular boxes represent the relays and the triangular figures represent the radios.

### 3.3.3 Peer-to-Peer Smart IEDS Communicating at PSE&G DA

The Public Service Electric and Gas (PSE&G) Company implemented a scheme designed to improve reliability of supply to its customers. This is called the Advanced Loop Scheme (ALS). The customers of PSE&G were increasing so the demand and the reliability was becoming an issue, which is the reason why PSE&G decided to consider ALS. ALS is built on the utility's standard distribution scheme that used a normally open point to separate two feeders [1].

The previous scheme of PSE&G was not having automatic reconfiguration and was relatively slow and resulted in unnecessary dips for customers when the link closed due to a fault**.** Using peer-to-peer communication, as illustrated in Figure 3.13, the ALS eliminated closing on a fault for a reconfiguration operation. The peer-to-peer communication provides supervision before switching can be done in order to ensure that there will be no switching on to a fault [1].

The scheme uses a "close before open" methodology in which the link is closed prior to sectionalizing taking place. Customers on un-faulted line sections end up not being exposed to an outage - due to a fault or switching taking place to isolate the faulted line section. PSE&G realised that communication improves protection performance and added more protective devices along the feeders to reduce the number of customers being affected by a fault on a particular part of the feeder per section [1].

The network where PSE& G implemented ALS is a 13 kV network and it serves about 3,000 customers divided into two sections, with each section serving approximately 1,500 customers. The issue was that the customers experienced prolonged outages when a fault occurs on one section. The setup was in such a way that during a fault condition 1500 customers were affected by an outage [1].

ALS targeted smaller customer groups by dividing a network into sections of 500 customers each. When a fault occurs in a section, only the customers in that affected section were interrupted. The other customers on the circuit did not experience any interruptions, due to the use of a high-speed selective relay using fibre optic communication that allows only the faulted section to be interrupted. The advanced technology identifies the fault, closes the feeder-tie or normally open point re-closer before other customers are impacted, and clears the fault in less than a second. This resulted in a reduction in the number and duration of customer outages [1].

Figure 3.13 demonstrates the PSE&G de-centralized DAS that uses Meshed radios and multi-drop communication to collect information from all of the relays and breakers, but the relays are making de-centralized decisions.

University of Cape Town

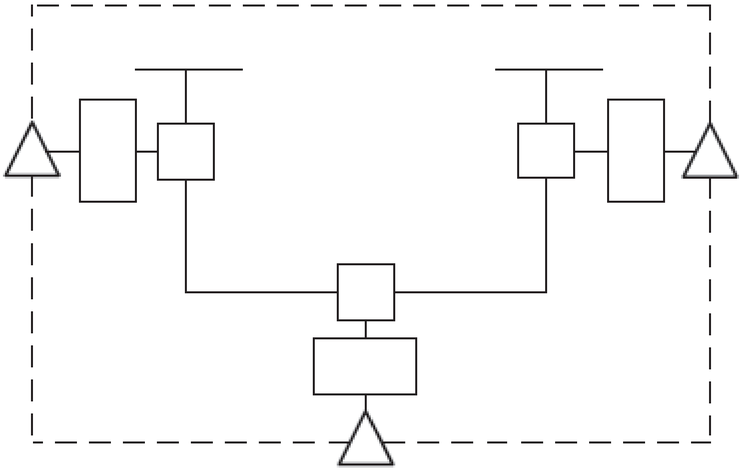


**Figure 3.13 peer- to-peer communications and decision making between the IEDs [1]**

In Figure 3.13, data in the peer–to-peer communication are also transmitted through multi-drop communication. They are transmitted from one set of breakers, relayed through the radio to the other set and vice versa. In this case the decisions are made in the relays. The square boxes represent the breakers, the rectangular boxes represent the relays and the triangular figures represent the radios.

### 3.3.4. Implementation of Substation Automation in South Africa

In Eskom Transmission South Africa, the implementation of Substation Automation (SA) is in order to collect data from IEDs in the substation to provide to the Network Control Centre (NCC). SA is also used for sending controls or commands, such as opening and closing of the circuit breakers from NCC to the substation Bay level. The implementation starts at Bay level, goes to Station level and ends at Remote control (Network Control Centre), where supervisory control and data acquisition (SCADA) are performed [31].

One of the substations where this is implemented is the Zeus 765 kV Transmission Station.

Table 3.2 provides the description and the functionality of the devices used in Eskom Transmission Station for the implementation of Substation Automation.

**Table 3.2 Devices used in Eskom Transmission Station for the implementation of Substation Automation [31]**

| Device Type | Description and Function |
|---|---|
| D400 - Gateway | A gateway is in essence a router. It is a device that is used to forward data packets along networks. It is connected to two or more LANs which have totally different addressing methods. It can also connect the LAN and its internet Service Provider (ISP) such as MTN service provider [29].<br>D400 gateway is a device that is used to transmit data between IEDs and the National Control Centre (NCC) using IEC61850 protocol. |
| Router | A router is described under a D400 gateway. |
| Gateway switch | A gateway is described under a D400 gateway |
| Station RTU | A Remote Terminal Unit (RTU) is a device that is used to transmit data |

University of Cape Town

| Device Type | Description and Function |
|---|---|
|  | between the relays and the National Control Centre (NCC). Unlike IEDs, they cannot communicate using IEC61850 protocol. |
| Substation Switch | A “switch” is a small hardware device that connects multiple computers together within one LAN. They are capable of examining data as they receive them, and to determine the source and destination Media Access Control (MAC) address of each data packet, as a result passing it to the relevant device or destination [30].<br>A substation switch is used as an interface between the backbone switch and the station RTU, Router and a D400 which is a gateway. |
| Backbone switch | A Backbone switch is described under the substation Switch. It is self-explanatory. It basically serves as a backbone of the substation whole network. It normally has a capacity of gigabits. For instance, it can transmit data in the substation at a rate of 1 or 10 Gbps. |
| Bay Switch | A bay switch is used as an interface between the IEDs at bay level. |
| IED | A protection device and it is used to protect the power system equipments. |

Figure 3.14 illustrates the typical network topology used for the implementation of SA for data collection from IEDs in the Transmission Station to NCC [31].

Figure 3.14 consists of IEDs, Bay switches, a substation switch, Backbone switches, a gateway switch, a router, station RTU and a D400. All these devices assist in the transmission of data in a network. When data (digital and analogue values) are transmitted from the substation, they first go through IEDs; these IEDs are connected to the breakers and the VTs and CTs. Then data go through the Bay Switch which is a device used to interconnect IEDs to each other and to the Backbone Switch. Data is then transmitted to the Substation Switch and Gateway switch. From the Gateway Switch, data are transmitted to the NCC. Another route to the NNC can also be through the station RTU. Data can also be transmitted as controls or commands from NCC to the IEDs, meaning data can be transmitted both ways.

University of Cape Town

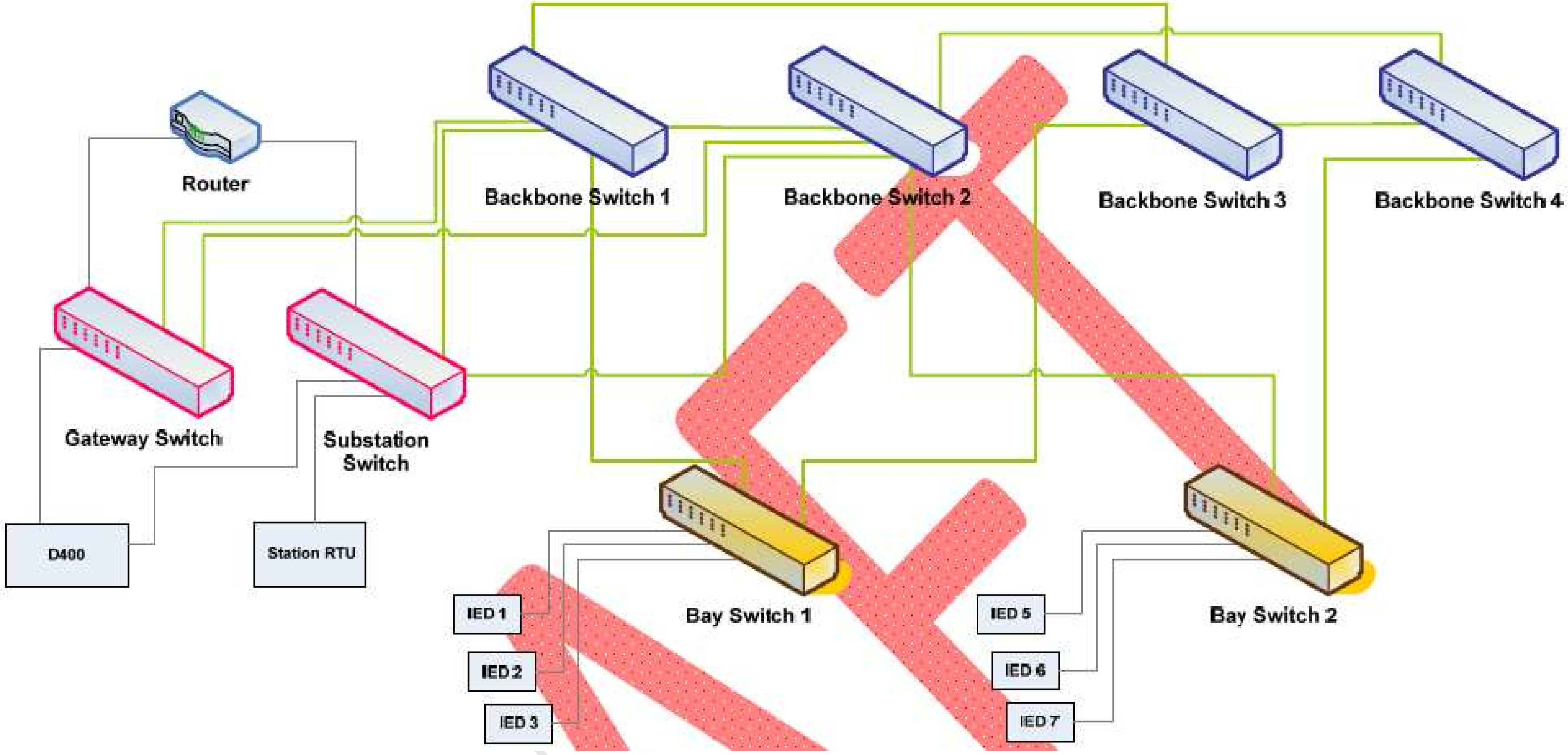


Figure 3.14 Typical Network topology for Transmission Station for data collection and control purposes [31]

# 4. Performance of GOOSE and Raw Data Messages.

Sidhu and Yin [5] simulated a 220 kV substation which consists of 2 transformer bays, 1 bus section bay and 6 feeder bays. The simulation process used is called the Opnet Model. The research was performed to evaluate the effect of background traffic on message tripping in a substation.

The OPNET Modeller creates IEC61850-based IED models, which means that data or messages that are generated (GOOSE and the Raw data) will pass the communication stack specified in IEC61850 standard [5].

According to the IEC61850-5, the message transmission time requirements for SA network must be guaranteed under any operating conditions and contingencies. The end-to-end messages (i.e. messages from the sending IED's Application layer to the receiving IED's Application layer) should not exceed a quarter of a cycle [5].

SAS consists of multiple components and each component serves multiple functions. In order to simulate the dynamic performance of an SAS network, IED models are constructed to represent the specific characteristics of an SAS network. The study of SAS network performance is separated into a process, bay and a station level (see Figure 4.1) [5].

of Cape Town

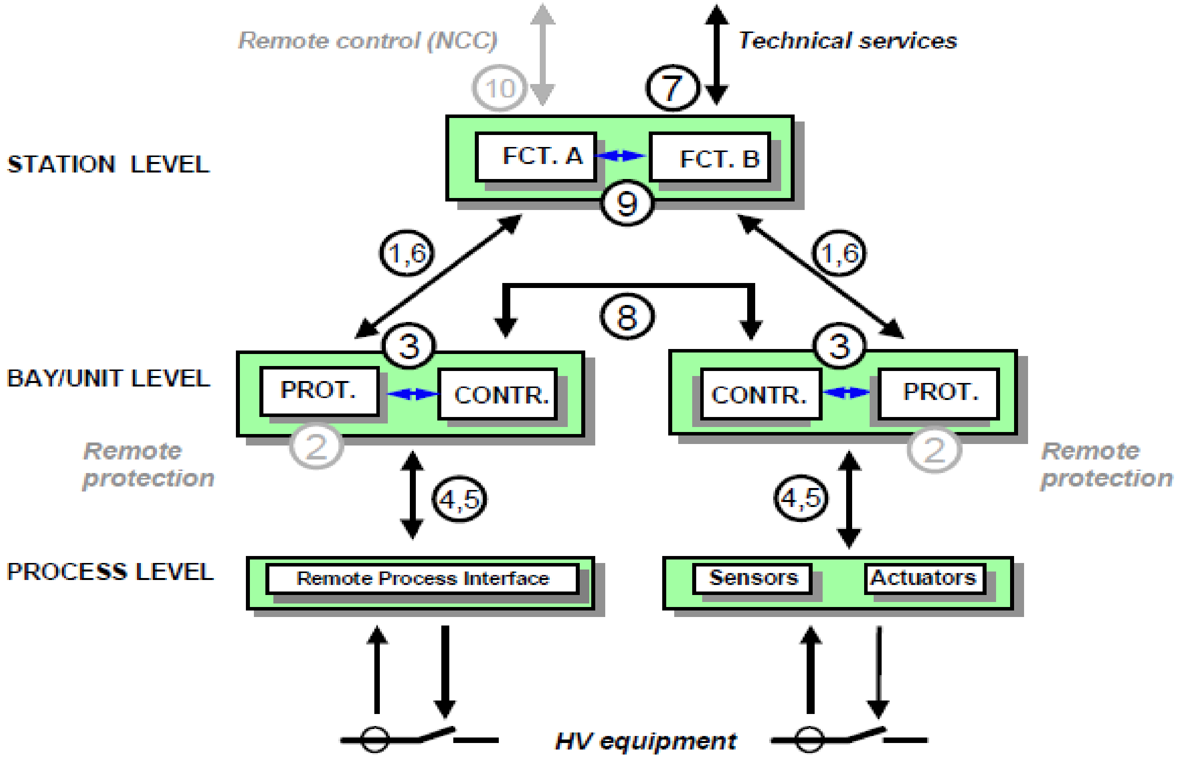


Figure 4.1 Interface model of a substation automation system [5]

Figure 4.1 indicates where different interfaces of a substation automation system exist, and their descriptions are specified as follows: [22:13-14]

**Interface (IF) 1**: protection-data exchange between bay and station level.

**IF3**: Data exchange within bay level.

**IF4**: CT and VT instantaneous data exchange (especially samples) between process and bay level.

**IF5**: Control-data exchange between process and bay level.

**IF6**: Control-data exchange between bay and station level.

**IF7**: Data exchange between substation (level) and a remote engineer's workplace.

**IF8**: Direct data exchange between the bays, especially for fast functions such as interlocking.

**IF9**: Data exchange within station level.

**IF10**: Control-data exchange between substation (devices) and a remote control centre (beyond the scope of this standard).

The OPNET Modeller constructs the models using an object-oriented modelling approach. In OPNET Modeller, Network devices like IEDs, switches and workstations are called node models. A node model consists of modules connected by packet streams or static wires. Each module is assigned to a process module to achieve the required behaviours. [5:1422-1489]

OPNET's process model uses a finite state machine (FSM) approach to support the implementation of protocols, resources, applications, algorithms, and queuing policies. As Sidhu and Yin [5] explain,

> FSM is a mathematical model used to design computer programs and digital logic circuits. It is conceived as an abstract machine that can be in one of a finite number of states. The machine is in only one state at a time; the state it is in at any given time is called the current state. It can change from one state to another when initiated by a triggering event or condition, this is called a transition.

The different IED models are discussed in the following paragraphs.

## 4.1 Modelling of IEDs

Sidhu and Yin [5] modelled three types of generic IEDs using the OPNET modeller: the breaker IED, merging unit (MU) IED, and protection & control (P&C) IED. The operation of these IEDs are described below.

University of Cape Town

- The MU IED first processes and combines the signals from the field Current Transformer (CT) and Voltage Transformer (VT). Then transmits the digital voltage and current output to the process bus (a high speed field Ethernet bus).
- The breaker IED monitors the state and condition of the circuit breaker. It receives the trip/close command from the P&C IEDs then sends it to the breaker. It also sends state change events to the corresponding P&C IEDs through the process Bus.
- The P&C IED is a universal device and it integrates the protection & control functionalities for the bay unit.

### 4.1.1 Categories of IEC61850 messages

As was stated earlier, modelling of IED follows the communication stack specified in IEC61850. The communication stack is illustrated in Figure 4.2 [5:1422-1489].

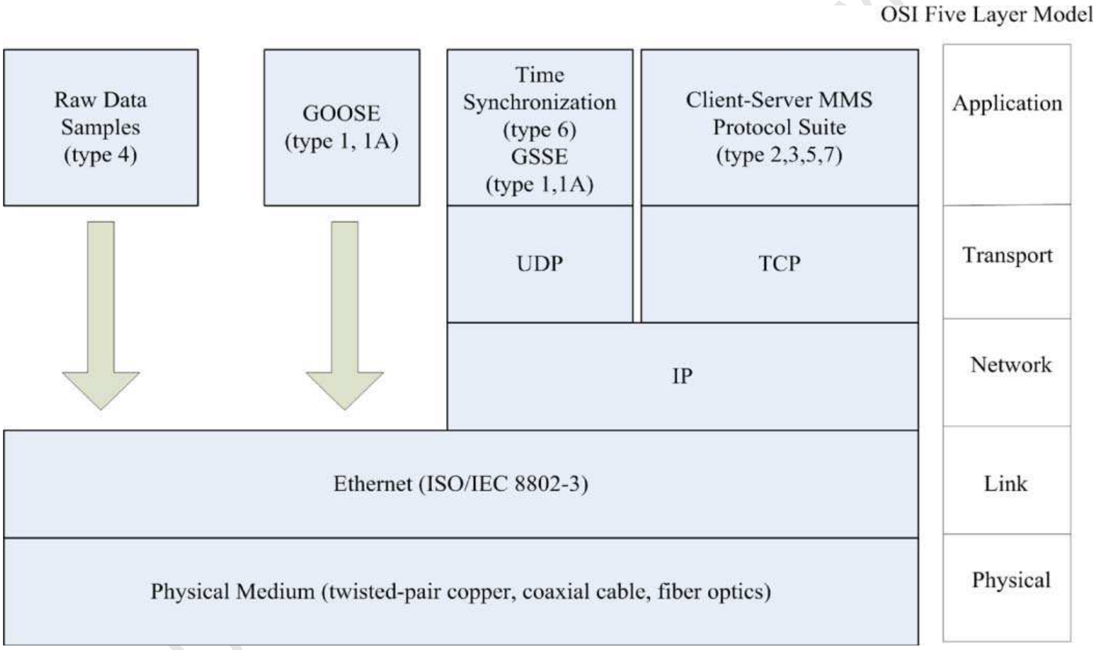


**Figure 4.2 IEC61850 communications stack [5]**

According to IEC61850-5 and IEC61850-8, messages are classified into 7 categories. The 7 types of messages are mapped into different communication stacks because of their performance requirements [5]. The following are the message requirements in accordance with the IEC61850 Communication stack:

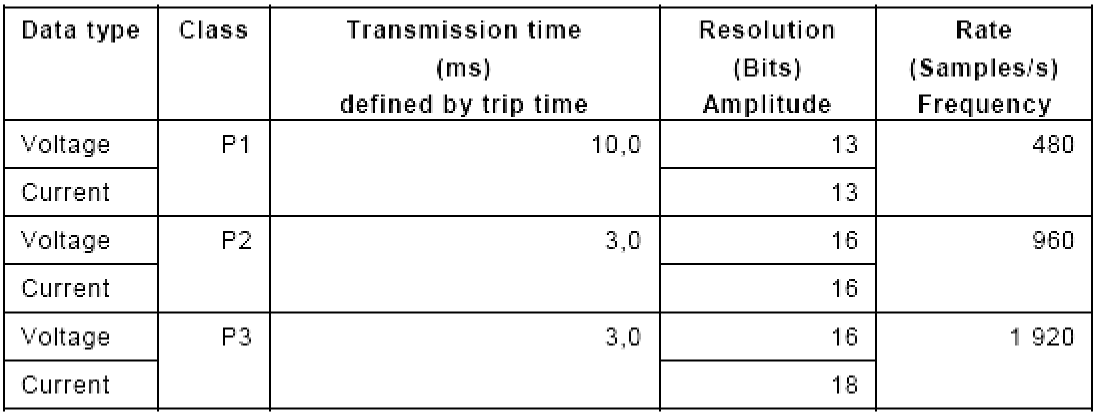
- GOOSE messages (type 1, 1A) and the Raw data samples (type 4) are time critical, and their transmission time is very important as they are used for protection purposes. These messages only occupy the lower–level of the IEC61850 communications stack, which are the Physical, Data Link, and Application layers. This gives the advantage of improved performance for real time messages through shorter Ethernet frames. The Ethernet frames of the Raw Samples values and GOOSE messages are shortened since the frames do not have the headers of other communication levels, such as the header of the Network layer - resulting in the reduction of the processing time of transmitted messages [5].

The type 1 messages contain commands or a message such as “Trip”, “Close”, “Reclose order”, “Start”, “Stop, “Block” and “Unblock”. The receiving IED will act immediately in accordance to the received message [5]. The type 1A message, “Trip”, is the most important message in the substation. Compared to all the other fast messages, these have more demanding requirements. The same performance may be requested for inter-locking, inter- tripping and logic discrimination between protection functions. Interlocking is a method of preventing undesired states in a state machine [26]. Inter-tripping is when a signal is sent to trip and reclose a remote circuit breaker irrespective of the remote protection sensing a fault or not [27].

For performance of class P1 messages, which applies typically for distribution bays or bays where low requirements are accepted, the message transmission time shall be half cycles, therefore 10 milliseconds (refer to Table 4.1). For performance of class P2 or P3 messages, which applies typically for transmission bays, the transmission time shall be less than a quarter of a cycle, therefore 3 milliseconds in this case (refer to Table 4.1) [5].

Type 4 messages include the output data from the digitizing transducer and digital instrument transformer. The type 4 data consist of continuous synchronous data from each IED; these data are incorporated with data from other IEDs [5].

**Table 4.1 Raw Data for Protection [5]**



| Data type | Class | Transmission time (ms) defined by trip time | Resolution (Bits) Amplitude | Rate (Samples/s) Frequency |
|---|---|---|---|---|
| Voltage | P1 | 10,0 | 13 | 480 |
| Current | | | 13 | |
| Voltage | P2 | 3,0 | 16 | 960 |
| Current | | | 16 | |
| Voltage | P3 | 3,0 | 16 | 1 920 |
| Current | | | 18 | |

- Type 2 messages are the event recording messages and are medium speed messages. When it comes to type 2 messages, the time at which the message originates is vital, as compared to its transmission time which is less critical. It is expected that the IEDs will have their own clocks and that the messages that are sent are time-tagged by the sender. These messages include a single measurement value such as the R.M.S value, which is calculated by the type 4 messages. The total transmission time for type 2 messages should be less than 100 milliseconds [5].

- Type 3 and Type 5 messages are both low speed messages. Type 3 messages include messages such as events records and alarms that require a time-tag. The total transmission time is required to be 500 milliseconds or more [5].

- Type 5 messages are used to transfer large file messages, such as settings files, and are mapped to MMS protocol suits which go through a Transmission Control Protocol/Internet Protocol (TCP/IP) stack above the Ethernet layer, Thus messages will go through the Physical, Data Link, IP, TCP and Application layers. As a result, an Ethernet frame of the settings file will be longer than that for GOOSE messages and sampled values. The transition time is required to be equal to 1 second or greater [5].

- Type 6 messages are time synchronisation and GSSE messages. The messages are broadcasted to all IEDs in a substation using User Datagram Protocol/Internet Protocol (UDP/IP). These types of messages are used to synchronise the internal clock of the IED for different purposes, e.g. correct time tagging of events or sample accuracy of Raw data [5].

  Generic Substation State Events (GSSE) messages are an extension of the event transfer mechanism in Utility Communication Architecture UCA2.0. As compared to the GOOSE messages, GSSE only report status changes and are transmitted directly over IEC/ISO 8802-2 and 8802-3 using a similar mechanism to GOOSE. GSSE is being progressively superseded by the use of GOOSE and support for it may eventually disappear [23].

- Type 7 messages are the command and access control messages and are used to transfer control order from the local and remote HMI where high security is required. These messages use Interface 7 and all the messages using this interface require access control (refer to Table 4.2) [5].

The messages can be transmitted in the substation using the different methods: polling, publishing, log or report. For example, Raw data samples are published by MU IEDs to P&C IEDs. GOOSE messages are published between P&C IEDs and breaker IEDs. Meter values or breaker status could be updated by polling or reporting of IEDs.

Table 4.2 summarise the kinds of SA messages that are transmitted through the IEC61850 communication stack.

**Table 4.2 SA messages transmitted through the IEC61850 communication stack [5]**

| Type | Name | Examples |
|---|---|---|
| 1a | Fast messages - Trip | Trips |
| 1b | Fast messages - Others | Commands, simple messages |
| 2 | Medium speed messages | Measurands , rms values |
| 3 | Low speed messages | Parameters, event recordings |
| 4 | Raw data messages | Output data from transducers and instrument transformers |
| 5 | File transfer functions | Large files |
| 6a | Time synchronism messages a | Time synchronism, station bus |
| 6b | Time synchronism messages b | Time synchronism, process bus |
| 7 | Command messages with access control | Commands from station HMI |

University of Cape Town

## 4.1.2 Modelling of Merging Unit IED

The modelling of the MU IED is based on IEC61850-9-1. The communication stack for the MU IED is very simple as it only consists of three layers of the OSI model. It contains an Application layer, Ethernet/Data Link layer, and Physical layer [5].

Figure 4.3 and 4.4 show the node model diagram for the MU IED bus topology and Star topology, respectively. The topologies are clearly defined under section 3.1 'Network Topologies'.

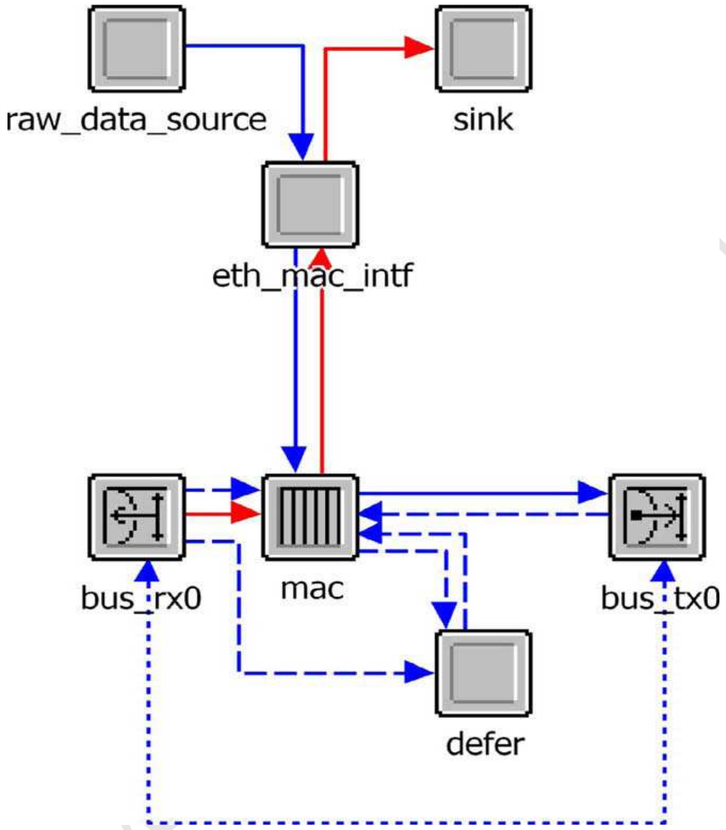


**Figure 4.3 The node model diagram for MU IED bus topology**

In Figure 4.3, the Raw data source module corresponds to the source for Raw data. The Raw data source module is on the Physical layer of the OSI model. bus_tx0 is a transmitting part of the node and bus_rx0 is a receiving part of the node.

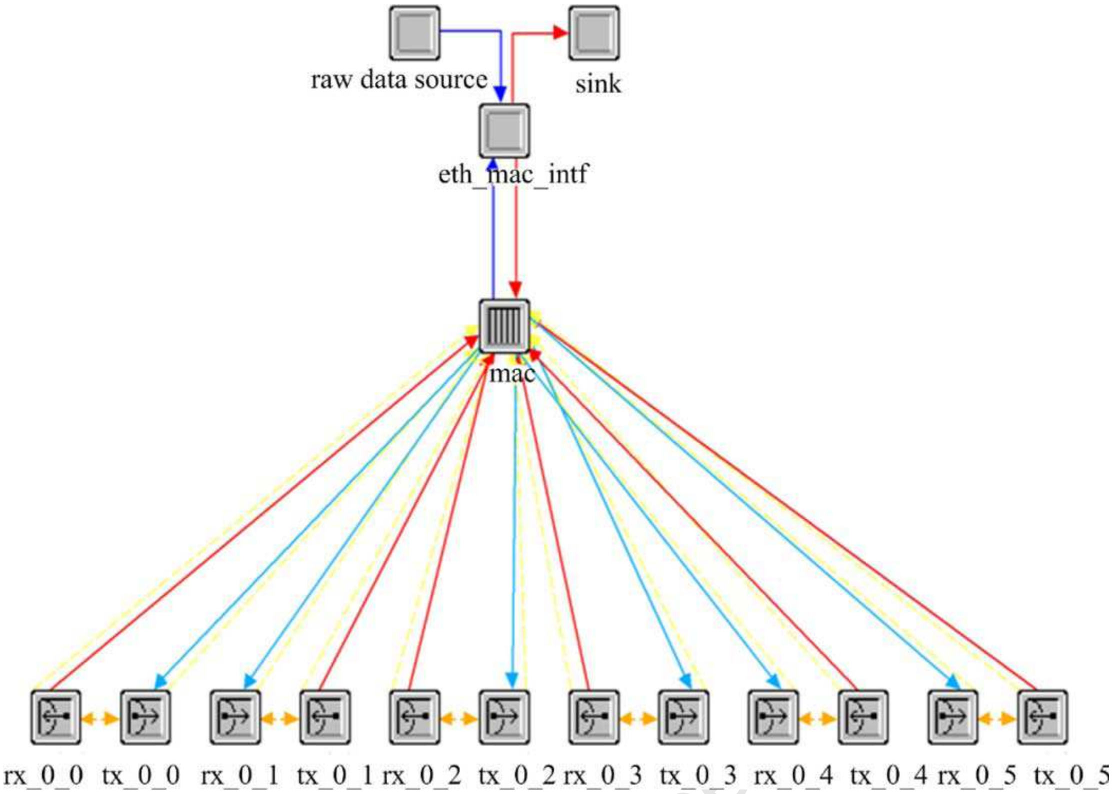


**Figure 4.4 The node model diagram for MU IED Star topology**

For the MU IED connected in a bus topology the P&C IEDs will also be connected in a bus topology. This can be done via a process bus, as illustrated in Figure 4.5. For the Star topology there are two ways to connect the MU IED to P&C IEDs. One way is to use a multiple point-to-point communications method, as shown in Figure 4.6, the other way is to connect the MU IEDs and P&C IEDs directly using an Ethernet switch, which only uses one communication port of the MU IED, as shown in Figure 4.7 [5].

The Application layer includes the Raw data source module which generates an Ether-type protocol data unit (PDU) as shown in Figure 4.4. This PDU can be data generated and transmitted by one device to another. The Ethernet (Data Link) layer consists of the eth_mac_intf, defer (for bus topology) and MAC modules.

The PDU contains an application protocol data unit (APDU) which may contain a number of Application Service Data Units (ASDUs) (refer to Figure 4.8). Each ASDU again contains four current values and four voltage values as specified in the standard [5].

The Physical layer allows connecting this IED to a process bus or multiple point-to-point bay devices using 10 Mbps, 100 Mbps, 1 Gbps or even a 10 Gbps link, depending on the type of transmitters and receivers this module uses [5].

University of Cape

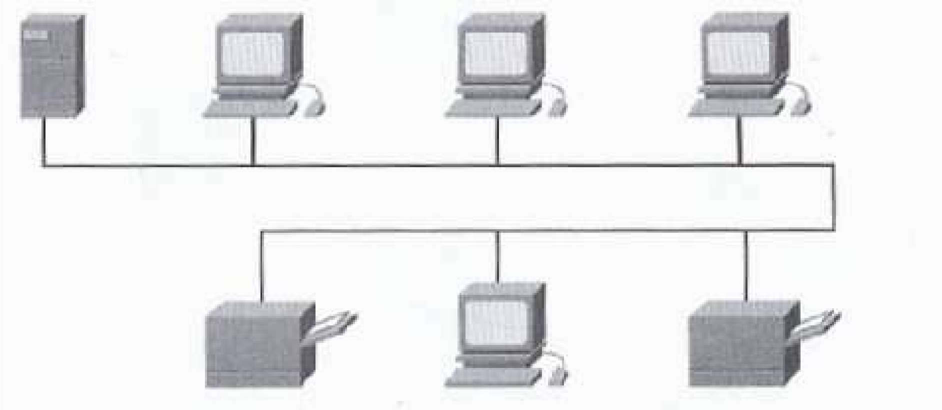


Figure 4.5 MU IED connected with the P&C IEDs through a process bus [7]

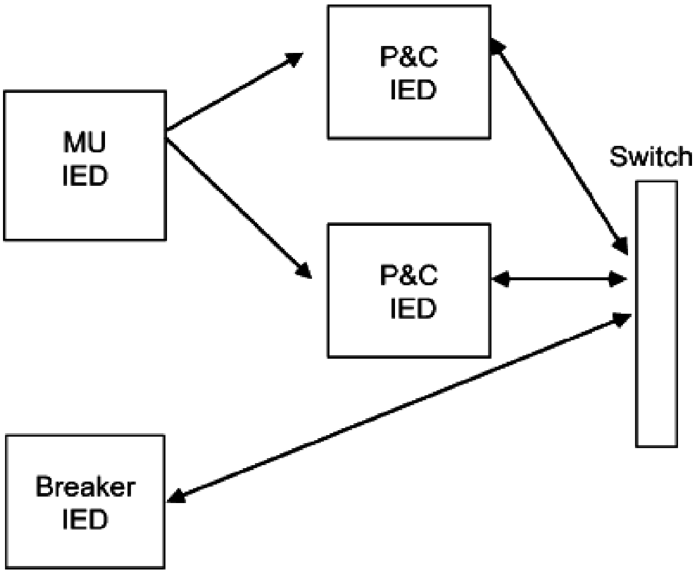


Figure 4.6 MU IED multiple point-to-point communication with P&C IEDs [5]

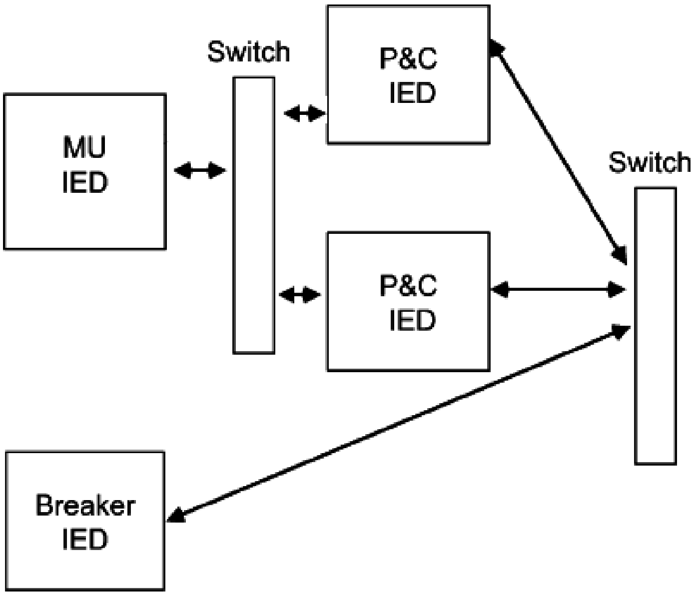


Figure 4.7 MU IED and P&C IEDs directly connected using an Ethernet switch [5]

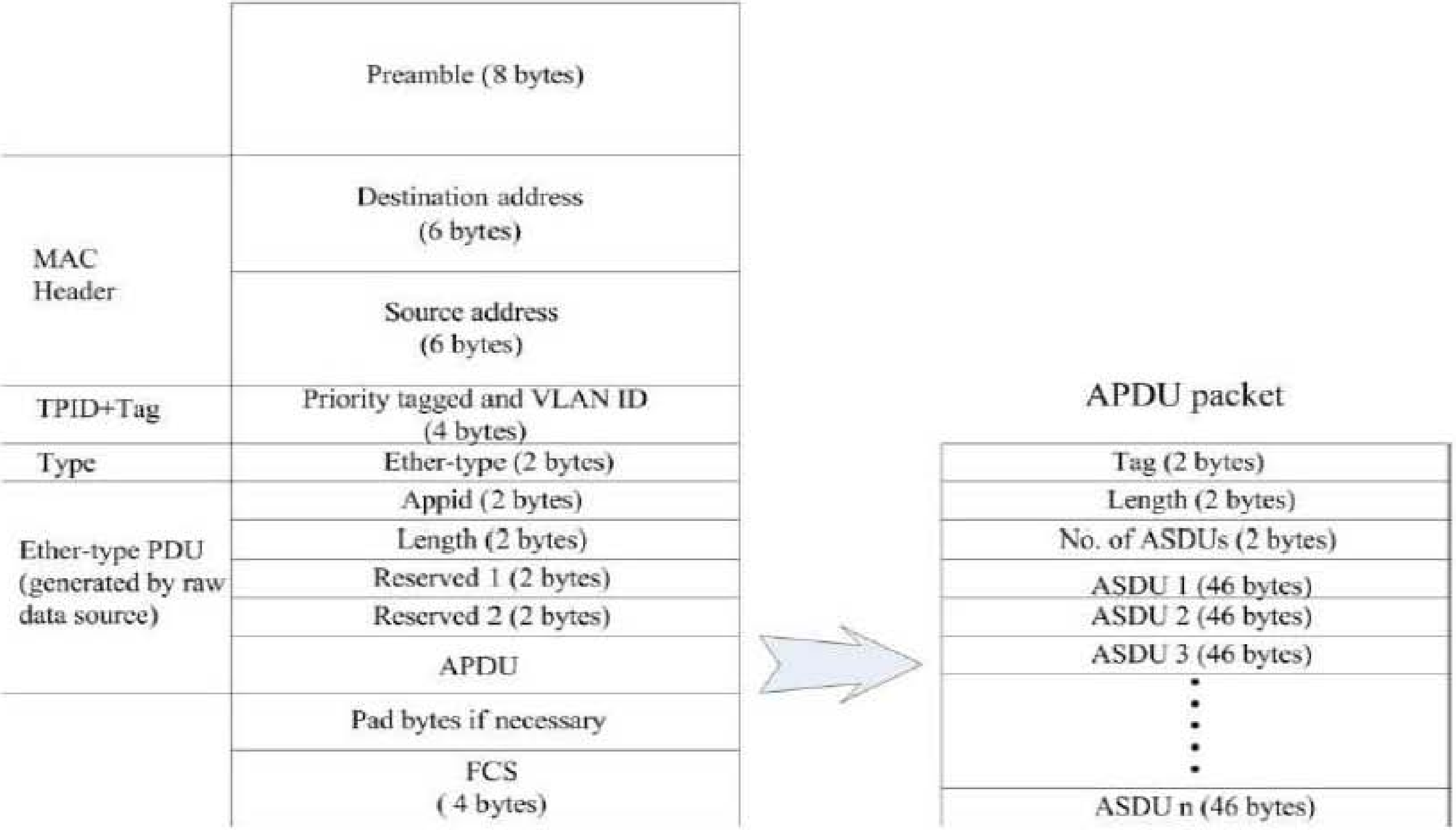


**Figure 4.8 Raw data sample Ethernet frame [5]**

### 4.1.3 Modelling of Circuit Breaker IED and Protection IED

The functionalities of a circuit breaker IED are to receive the trip signal, calculate the end-to-end (ETE) delay and send a multicast GOOSE event to protection IEDs and station PC to notify them of the event that has just taken place. The ETE delay is the time between the creation of the message at the Application layer of the sending IED and the arrival of the message at the receiving IED's Application layer. [5]

Like the sampled values, the GOOSE/GSSE messages are time critical. They should be tagged with high priority. The GOOSE messages have direct access to the low-level Ethernet layer. However, unlike the MU IEDs, which just send the sample value to the P&C IEDs and receive nothing from the P&C IEDs, the breaker IED has to exchange messages with the protection IEDs and station PC. The messages are sent to and received from P&C IEDs and the Station PC. Therefore breakers IEDs are modelled to support client-server communication, which means they are either a client or server at one point of their communication to the P&C IEDs or the Station PC. Figures 4.9 and 4.10 illustrate the breaker IED models used for bus topology and Star topology respectively.

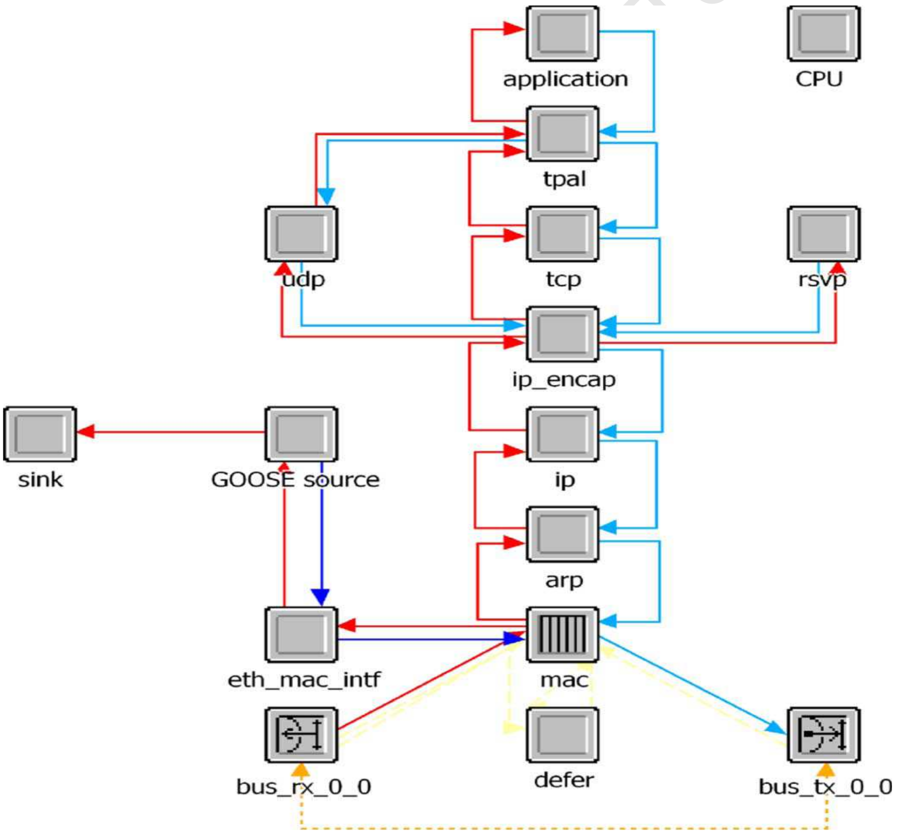


**Figure 4.9 The node model diagram for breaker IED bus topology**

In Figure 4.9 the bus topology of the breaker IED is indicated. Figure 4.9 shows bus_rx_0_0 and bus_tx_0_0 representing the receiving bus and transmitting bus respectively. The layer where the bus_rx_0_0 and bus _tx_0_0 exist is the Physical layer.

When data are transmitted, they start from the Application layer of the transmitting bus and then go through the Transport layer, which is either the Datagram Protocol (UDP) layer or the Transmission Control Protocol (TCP) layer. Data will then go through the Network layer, which is the Internet Protocol (IP) layer. From the IP layer, data will go through the Data Link layer, which is the Media Access Control (MAC) Address layer, then the Physical layer of the Transmitting bus. Data will then be transmitted through to the Physical layer of the Receiving bus all the way to the Application layer of the Receiving bus through the MAC, IP and TCP/UDP layers [5].

The route is different when it comes to the GOOSE messages though. GOOSE messages go through the Application, Data Link/MAC and the Physical layer only. In this case, the GOOSE messages are created by the OPNET modeller and start from the GOOSE source then proceed to the MAC layer (blue arrows) and lastly to the Physical layer. The GOOSE messages are also transmitted back to the GOOSE source and into the sink where the GOOSE messages are received and thrown out (red arrows from MAC to sink) [5].

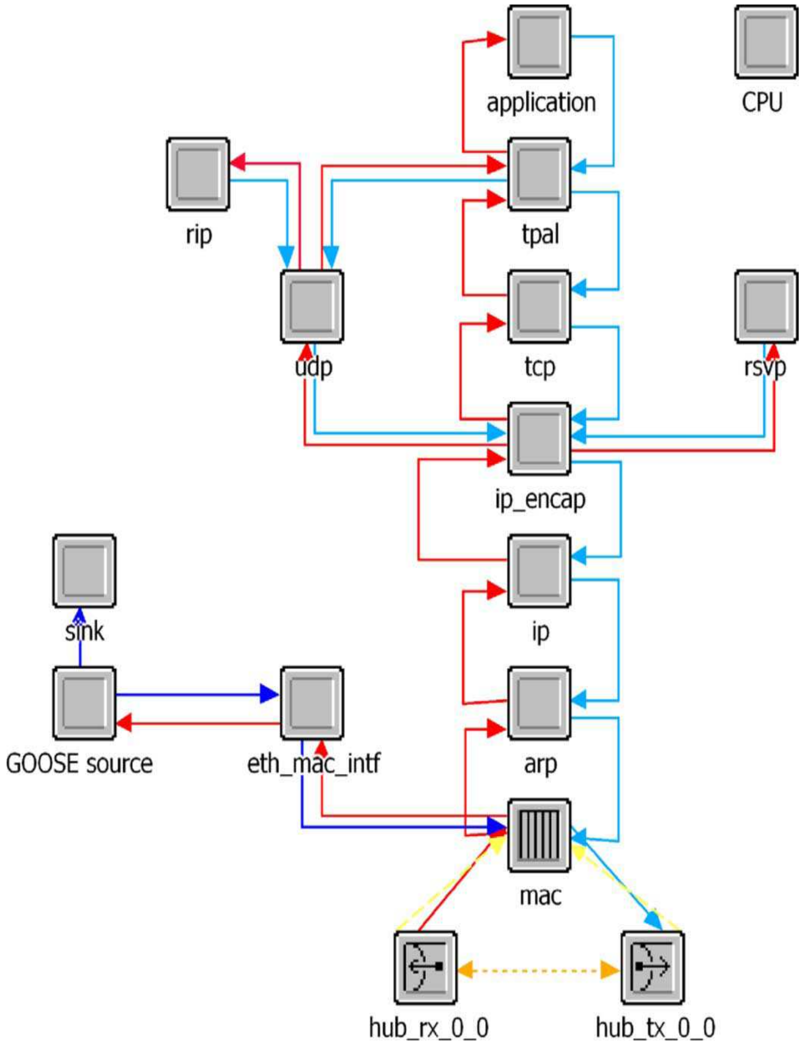


**Figure 4.10 The node model diagram for Breaker IED Star topology [5]**

In Figure 4.10 the Star topology of the breaker IED is indicated. Figure 4.10 shows hub_rx_0_0 and hub_tx_0_0 representing the receiving hub and transmitting hub respectively. The process for Breaker IED bus topology is similar to the Breaker IED Star topology, except with the communication interfaces. Compared to the Breaker IED bus topology, the Breaker IED Star topology uses the hub (a communication device) instead of a bus (single cable). The GOOSE messages are created by the OPNET modeller and start from the GOOSE source, then proceed to the MAC layer (blue arrows) and lastly to the Physical layer. The GOOSE messages are also transmitted back to the GOOSE source and into the sink where the GOOSE messages are received and thrown out (red arrows from MAC to sink).

The P&C IED is similar to the breaker IED except it has two bus communication ports for connecting to the process bus and station bus in the bus topology, and for connecting MU IED and Breaker IED in the Star topology, the node model for the P&C IEDs are illustrated in Figures 4.11 and 4.12. In the case of no process bus or MU IEDs, the ports are left unused [5].

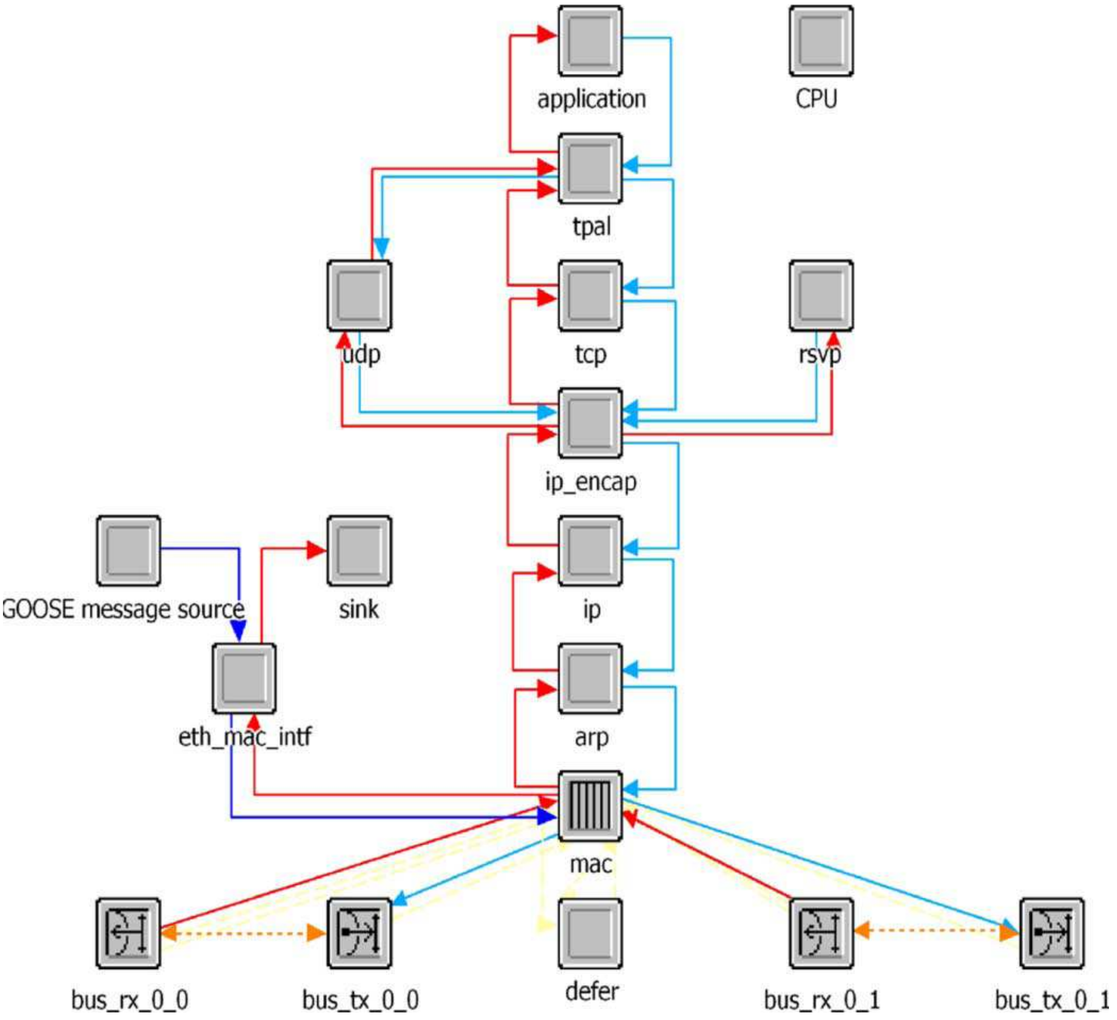


**Figure 4.11 P&C IED model for bus topology [5]**

In Figure 4.11 the bus topology of the P&C IED is indicated. Figure 4.11 shows bus_rx_0_0 and bus_tx_0_0, which represent the first receiving bus and transmitting bus respectively, and bus_rx_0_1 and bus_tx_0_1, which represent the second receiving bus and transmitting bus respectively. The process for P&C IED bus topology is similar to the Breaker IED bus topology. Compared to the Breaker IED bus topology, the P&C IED bus topology uses two bus nodes to communicate with both the Breaker IED and the MU IED, while the Breaker IED bus topology uses one bus node

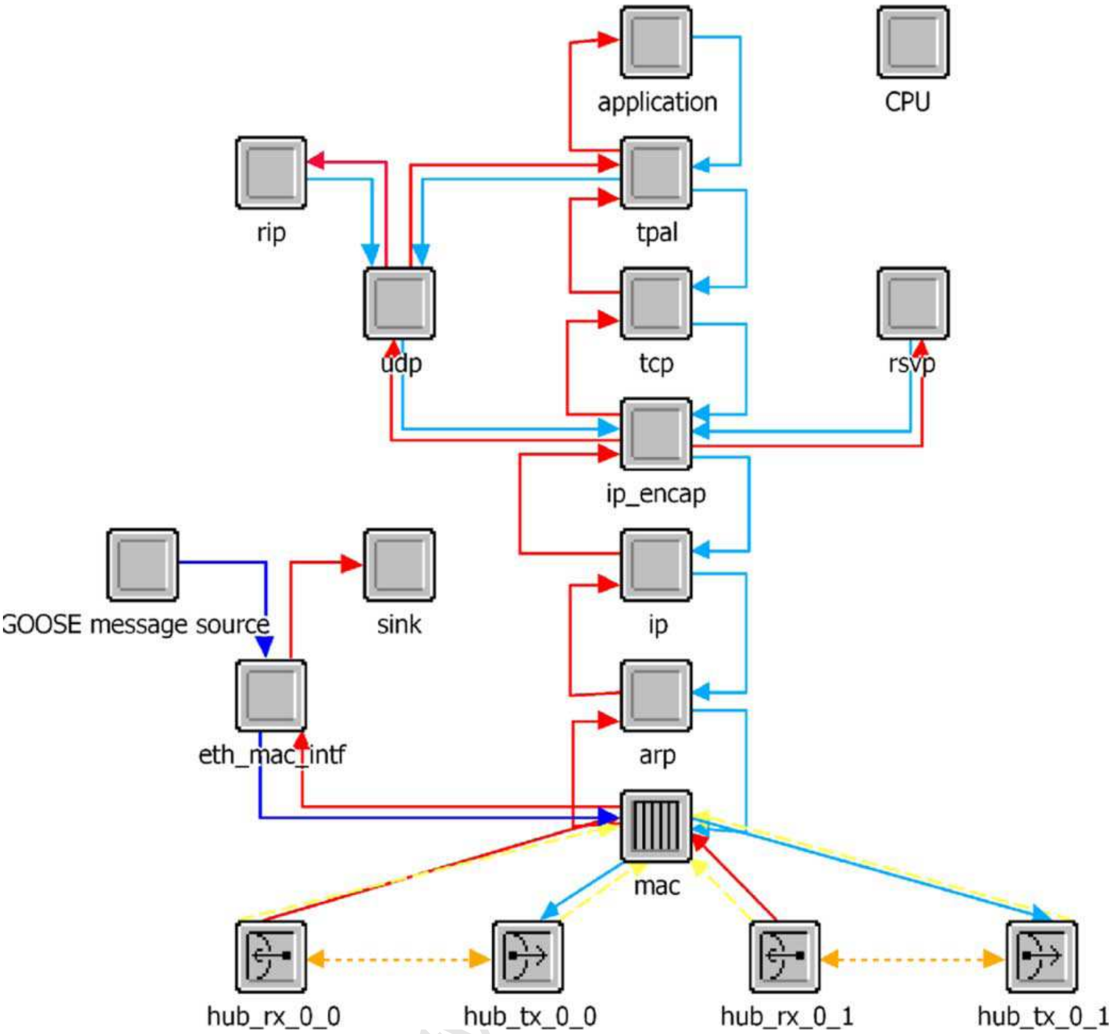


**Figure 4.12 P&C IED model for Star topology [5]**

University

In Figure 4.12 the Star topology of the P&C IED is indicated. Figure 4.12 shows hub_rx_0_0 and hub_tx_0_0, which represent the first receiving hub and transmitting hub respectively, and hub_rx_0_1 and hub_tx_0_1, which represent the second receiving hub and transmitting hub respectively. The process for P&C IED Star topology is similar to the Breaker IED Star topology. Compared to the Breaker IED Star topology, the P&C IED bus topology uses two hub nodes to in order to communicate with both the Breaker IED and the MU IED while the Breaker IED bus topology uses one hub node

The P&C IEDs can be configured to generate background traffic flow to station server or station PC. When the fault occurs, the P&C IED will send a trip message at a specified time. The trip message will be multicast to the corresponding or prescribed breaker IEDs. For the IEDs to respond correctly, the user needs to configure the addresses, the source address, destination address, multicast group addresses, and other parameters according to the simulation requirements [5].

When it comes to network performance, one should consider the ETE delay for time critical messages to be a key statistic that reflects network performance [5]. For example, according to an IEC61850 standard, the trip message ETE delay in the distribution bays should not be greater than half a cycle which is calculated to be 10 milliseconds - if it is greater than 10 milliseconds it should be considered slow [5].

## 4.2 Sample SAS Construction and Performance Evaluation

As is stated above, the OPNET model simulated a 220 kV substation which consists of 2 transformer bays, 1 bus section bay and 6 feeder bays. Figure 4.13 illustrates the simulation process used in the OPNET model.

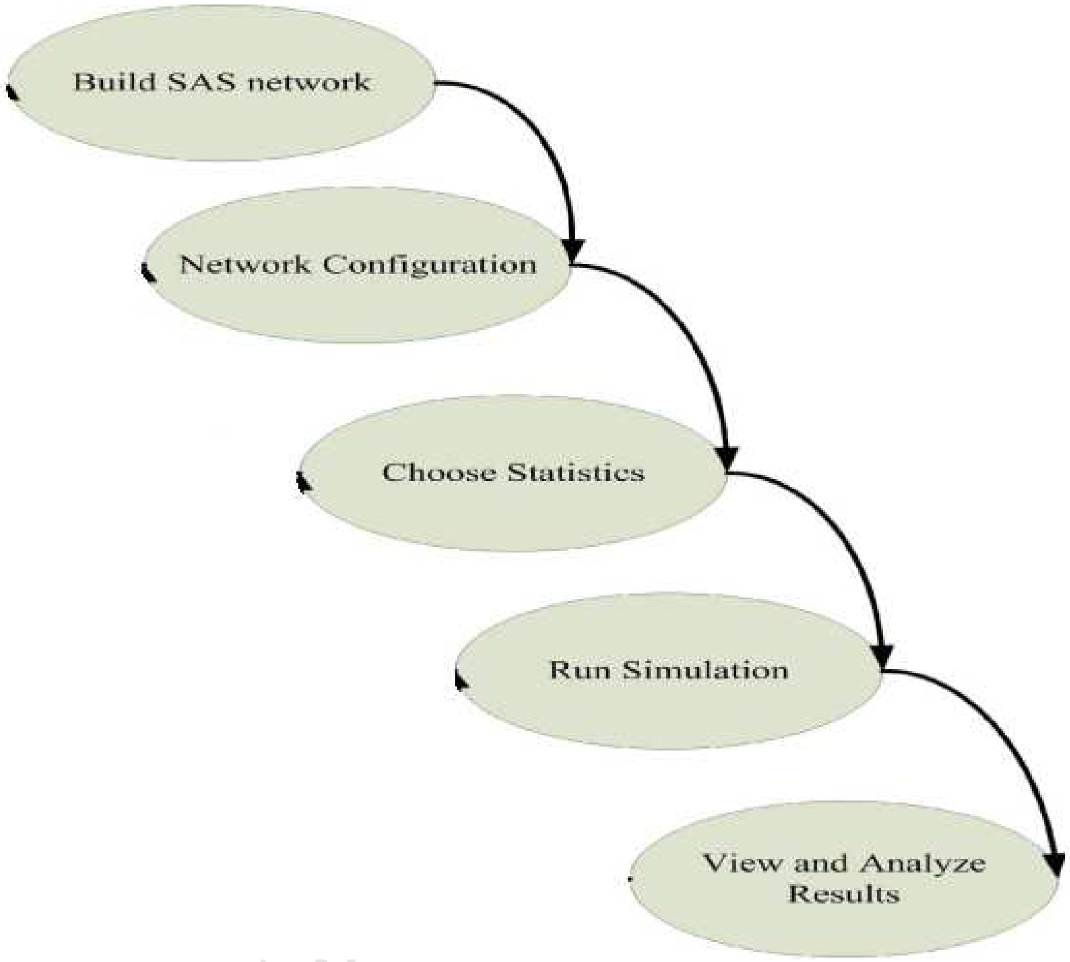


**Figure 4.13 A simulation process used in OPNET model [5]**

Figure 4.13 the simulation process is illustrated. The simulation process starts by building a Substation Automation (SA) System, followed by Network configuration and the choosing of statistics which entails choosing of Buses and feeders, etc. Then the simulation is run, and the results are viewed and analysed.

Figure 4.14 illustrates the single line diagram of the 220 kV network that was simulated.

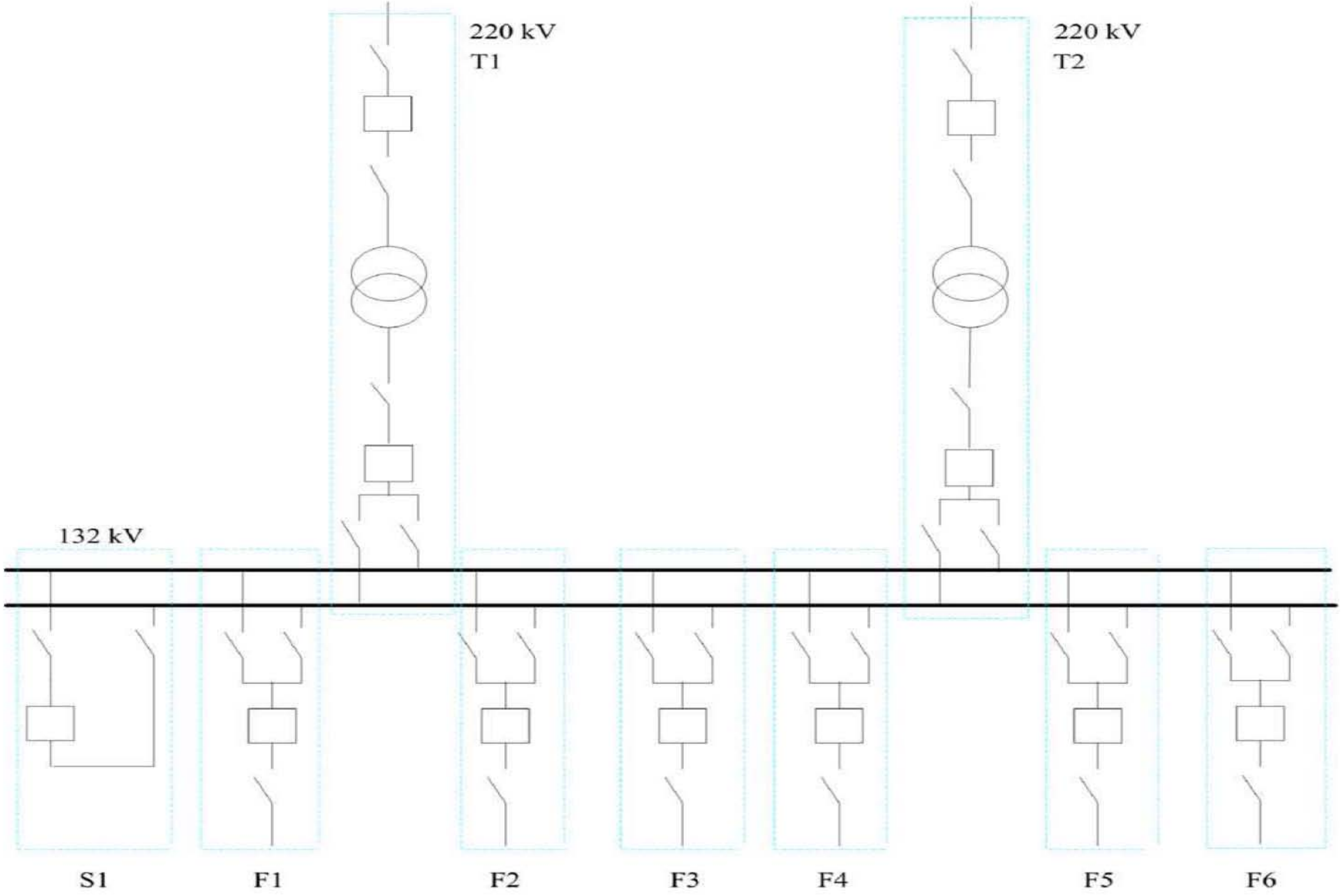


Figure 4.14 Single line diagram of the 220 kV network [5]

A switched Ethernet network has been constructed using the SAS performance research platform. Each feeder bay is modelled into a subnet which contains one breaker IED, one MU IED, two P&C IEDs and one Ethernet switch. Each transformer bay is modelled into a subnet which contains two breaker IEDs, a combined MU IED (each packet contains two ASDUs), two P&C IEDs and one Ethernet switch. The bus section bay contains one breaker IED, one MU IED, one P&C IED and one Ethernet switch.

The two basic topologies which were tested and analysed in this research are the bus and Ring topologies. Figure 4.15 is a Star topology and it is highlighting subnet of transformer 1 (T1) bay and Feeder 1 (F1).

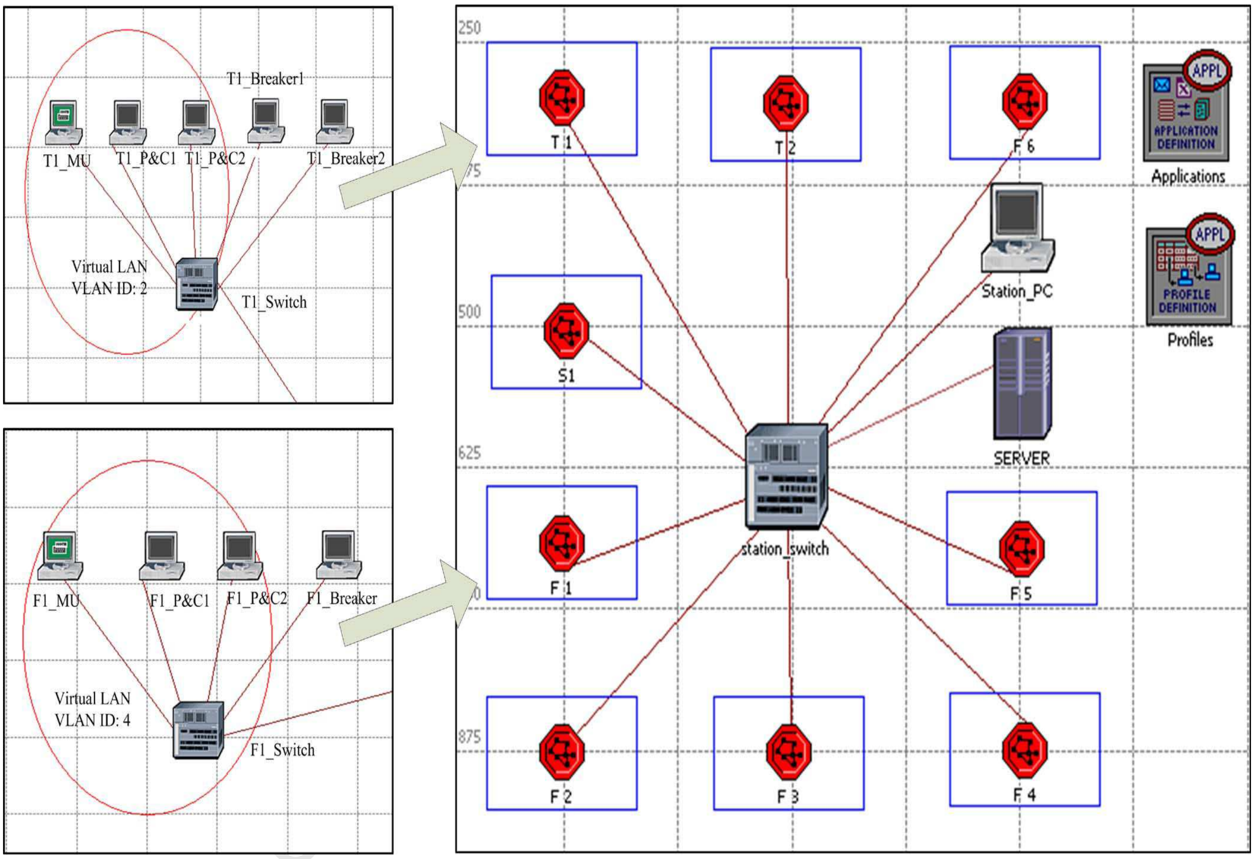


**Figure 4.15 The simulation of 220 kV on the project editor on OPNET model [5]**

Each MU IED is sending Raw data (type 4) messages to corresponding P&C IEDs at a specified sampling rate when the simulation starts. Each APDU generated by the transformer MU IED contains two ASDUs because it contains two datasets of current and voltage. The message size for the transformer bay is 98 bytes, which is the two 46 bytes of the APDU, two bytes of APDU and two bytes of length. Each APDU of the feeder bay contains one ASDU and the message size is 52 bytes. Each Raw data message is sent once to a network. VLAN is used to limit the broadcast domain into the bay level [5].

It is assumed that the fault causes two P&C IEDs in F1 and T2 bays to send trip messages to the corresponding breaker IEDs. In order to observe the effect of background traffic to trip messages, one of the T2 bay P&C IEDs sends trip messages continuously to its corresponding breakers and the breaker in S1 bay. The message size is 16 bytes. Each trip message is sent four times to ensure correct delivery of message to the breakers. The breaker IEDs also report a GSSE message to corresponding P&C IEDs. The message size is 16 bytes as well [5].

All P&C IEDs and breaker IEDs are sending updated meter values or breaker status to the station server at medium speed since these messages are type 2 messages. These messages are basically the r.m.s values of voltage and current and circuit breaker status. The message size is set to 32 bytes. Each message is sent once to the network, since the messages are not very critical like a trip signal or the raw sampled values [5].

All P&C IEDs are also randomly transferring files to the station server, this to introduce traffic in the background. Transmitting the critical messages and having the P&C IEDs transferring data other than trip signals at the same time can affect the ETE transmission time of trip signals [5].

Figure 4.16 illustrates the influence of data transfer on the ETE delay of Raw data sample (4800 samples/s) at the LAN speed of 10 Mbps at one instant and 100 Mbps at the other instant [5].

Figure 4.16 consists of four graphs, the blue and red graphs represent the Raw data that are sent into the network at 10 Mbps at the same time with rest of the data (randomly transferring files to station server). The difference between the blue and the red graph is that, the Raw data message in the blue graph is not tagged with the priority tagging as compared to the data in the red graph which is tagged with priority tagging. Data that are tagged with priority tagging receive preference in the communication queue; if there are other data that are transmitted at the same time, the tagged messages will go through or send off to the destination first before the rest of data that is not tagged.

University of Cape Town

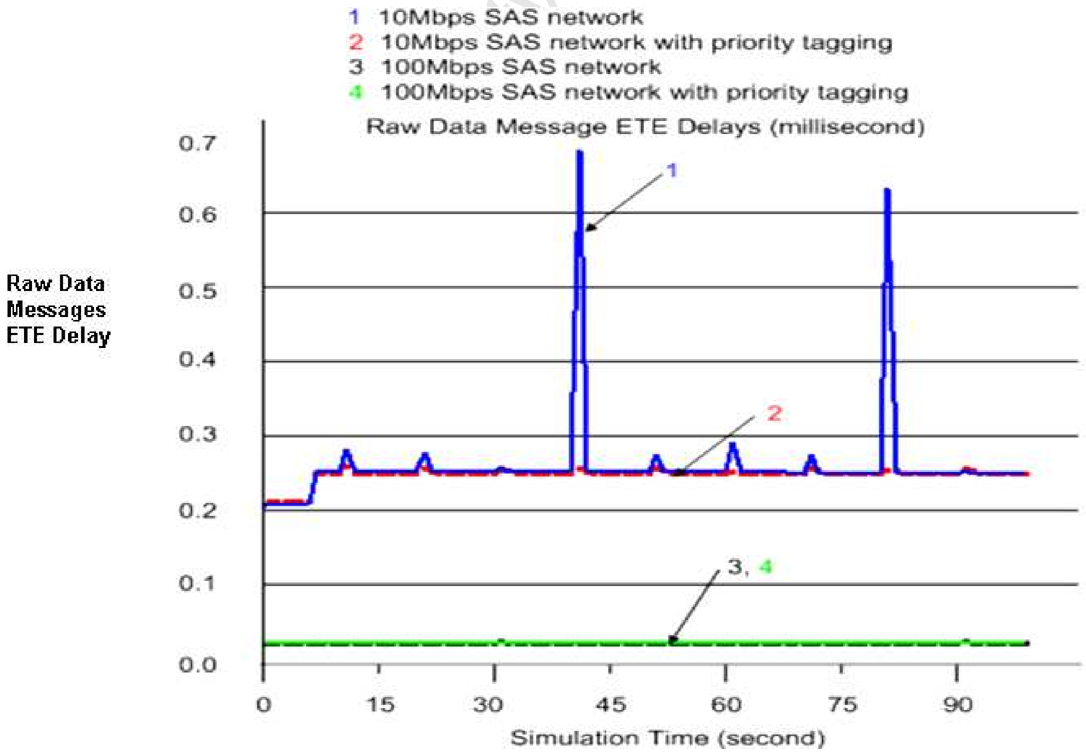
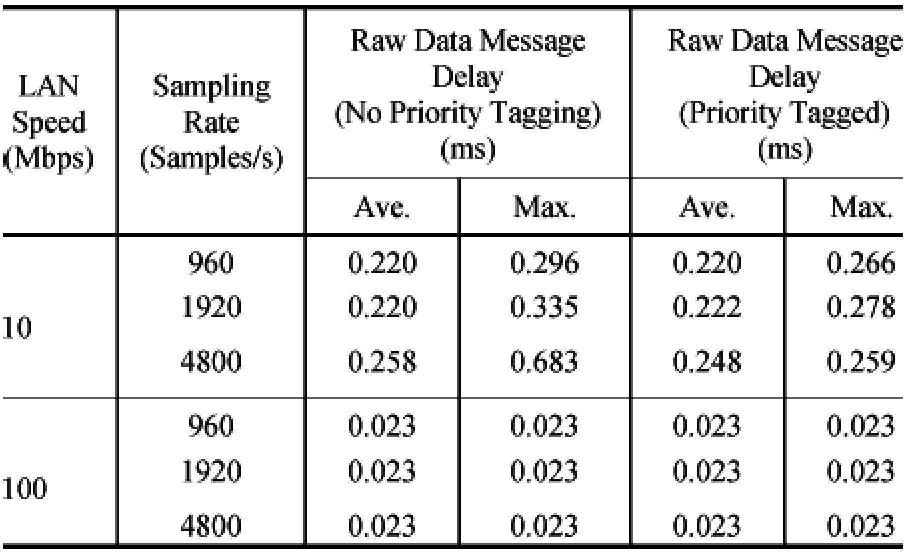


**Figure 4.16 ETE delay of Raw data sample at sample rate of 4800 samples per seconds with file transfer in a network. [5]**

The x-axis of the graphs represents the simulation time in seconds and the y-axis represents the Raw Data messages ETE delay, which is the time it takes for the message to leave the Application layer of the sending IED to the receiving IED. As it can be observed from Figure 4.16, priority tagging did make a difference in the ETE transmission time to Raw data message in the red graph. Table 4.3 illustrates that the maximum message delay due to file downloading is 0.683 milliseconds. It can also be observed that priority tagging has reduced the ETE delay by 0.259 milliseconds on the 10 Mbps LAN network [5].

The green and black graphs represent the Raw data that is sent into a network at 100 Mbps per seconds at the same time with the traffic. The black graph is not tagged with the priority tagging as compared to the data in the green graph which is tagged with priority tagging. It can be observed that when that data are sent into a network at 100 Mbps per seconds, there is not much difference between the ETE message transmission for the data message with priority tagging and the one without priority tagging. This simply means that with fast Ethernet at 100 Mbps, priority tagging, large file transfer has almost no effect on ETE delay of Raw data sample messages [5].

**Table 4. 3 RAW DATA MESSAGE ETE DELAYS [5]**

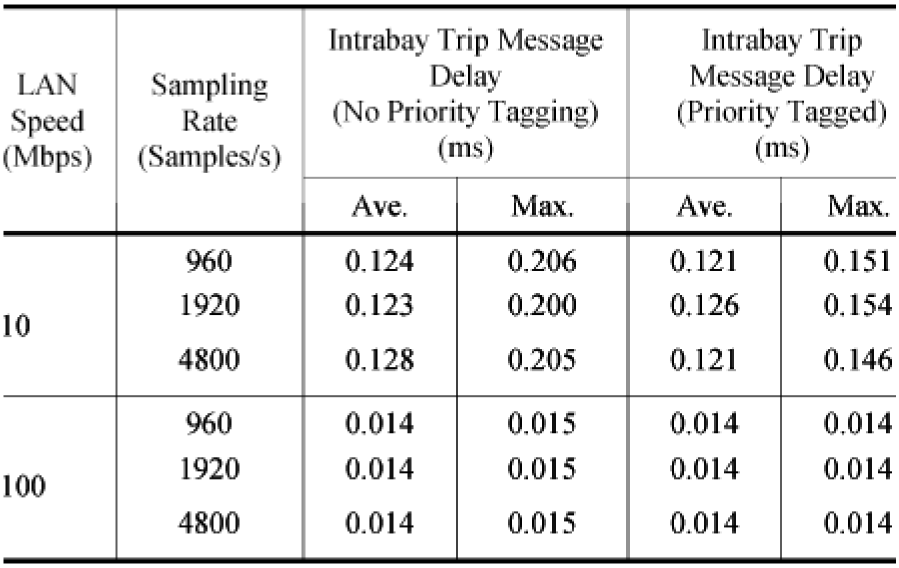


| LAN Speed (Mbps) | Sampling Rate (Samples/s) | Raw Data Message Delay (No Priority Tagging) (ms) | | Raw Data Message Delay (Priority Tagged) (ms) | |
|---|---|---|---|---|---|
| | | Ave. | Max. | Ave. | Max. |
| 10 | 960 | 0.220 | 0.296 | 0.220 | 0.266 |
| | 1920 | 0.220 | 0.335 | 0.222 | 0.278 |
| | 4800 | 0.258 | 0.683 | 0.248 | 0.259 |
| 100 | 960 | 0.023 | 0.023 | 0.023 | 0.023 |
| | 1920 | 0.023 | 0.023 | 0.023 | 0.023 |
| | 4800 | 0.023 | 0.023 | 0.023 | 0.023 |

Table 4.4 illustrates that the maximum message delay on the intra bay (message between IEDs of the same bay) trip messages due to file downloading on the 10 Mbps network is 0.205 milliseconds. It can also be observed that priority tagging has reduced the ETE delay by 0.145 milliseconds on the 10 Mbps LAN network.

When that data are sent into a network at 100 Mbps, the time difference between the message with no priority tagging and the message with priority tagging is 0.001 milliseconds.

Table 4.4 Intra- bay trip – messages ETE delay [5]

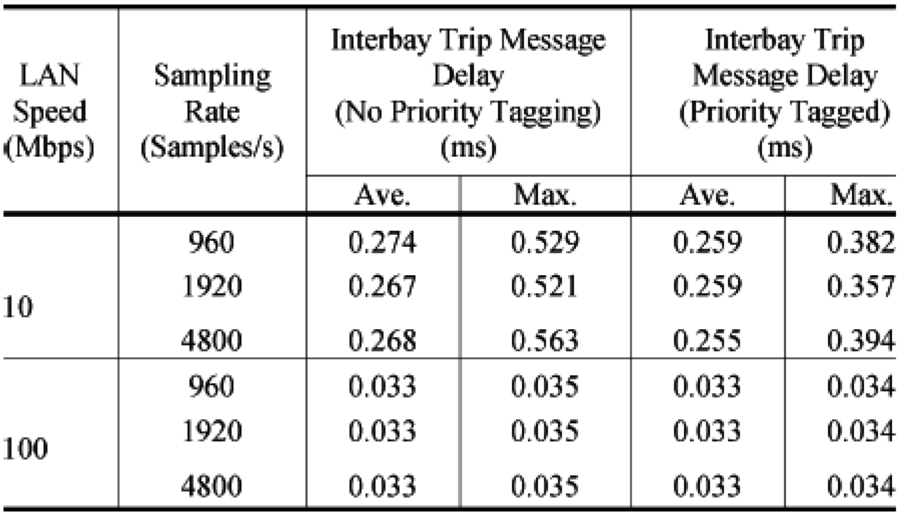


| LAN Speed (Mbps) | Sampling Rate (Samples/s) | Intrabay Trip Message Delay (No Priority Tagging) (ms) | | Intrabay Trip Message Delay (Priority Tagged) (ms) | |
|---|---|---|---|---|---|
| | | Ave. | Max. | Ave. | Max. |
| 10 | 960 | 0.124 | 0.206 | 0.121 | 0.151 |
| | 1920 | 0.123 | 0.200 | 0.126 | 0.154 |
| | 4800 | 0.128 | 0.205 | 0.121 | 0.146 |
| 100 | 960 | 0.014 | 0.015 | 0.014 | 0.014 |
| | 1920 | 0.014 | 0.015 | 0.014 | 0.014 |
| | 4800 | 0.014 | 0.015 | 0.014 | 0.014 |

Table 4.5 illustrates that the maximum message delay on the inter bay (message between IEDs from different/ separate bays) trip messages due to file downloading is 0.563 milliseconds. It can also be observed that priority tagging has reduced the ETE delay by 0.394 milliseconds on the 10 Mbps LAN network.

When that data are sent into a network at 100 Mbps network, the time difference between the message with no priority tagging and the message with priority tagging is still 0.001 milliseconds, the same as the intra bay trip ETE delay.

Table 4.5 Inter-bay trip messages ETE delay [5]



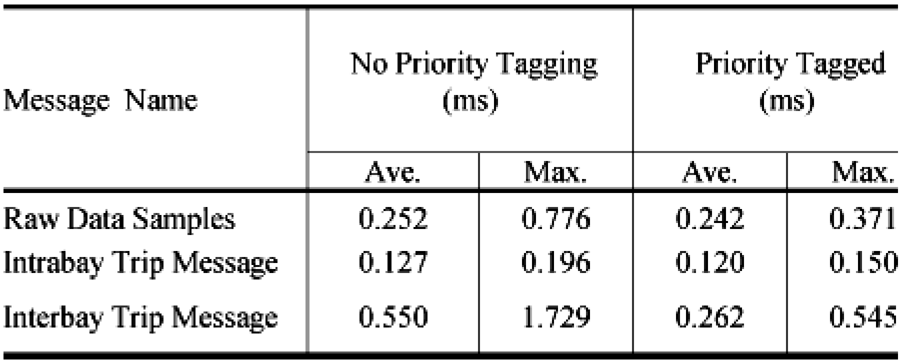
| LAN Speed (Mbps) | Sampling Rate (Samples/s) | Interbay Trip Message Delay (No Priority Tagging) (ms) | | Interbay Trip Message Delay (Priority Tagged) (ms) | |
|---|---|---|---|---|---|
| | | Ave. | Max. | Ave. | Max. |
| 10 | 960 | 0.274 | 0.529 | 0.259 | 0.382 |
| | 1920 | 0.267 | 0.521 | 0.259 | 0.357 |
| | 4800 | 0.268 | 0.563 | 0.255 | 0.394 |
| 100 | 960 | 0.033 | 0.035 | 0.033 | 0.034 |
| | 1920 | 0.033 | 0.035 | 0.033 | 0.034 |
| | 4800 | 0.033 | 0.035 | 0.033 | 0.034 |

Tables 4.3 – 4.5 show the general comparisons of network simulation results under different Raw data sampling rates and the LAN speeds of 10 Mbps and 100 Mbps. From the results, it can be observed that this network architecture satisfies the substation message performance requirements, even under the condition of 10 Mbps link speed without priority tagging. Priority tagging provided almost no performance improvement at 100 Mbps LAN because of the relatively light network traffic. However, the Star topology is

expensive as each of its bay units has an Ethernet switch and altogether ten Ethernet switches are used [5].

For the same substation, an economic network topology with only three switches can be and is simulated at 10 Mbps link speed. Under this topology the transformer switch connects two transformer bays, the feeder switch connects six feeder bays and the bus section bay, and the station switch stays the same. The substation message delay at a Raw data sampling rate of 4800 Hz is shown in Table 4.6 [10].

**Table 4.6 Economic Configuration – messages ETE delay [5]**



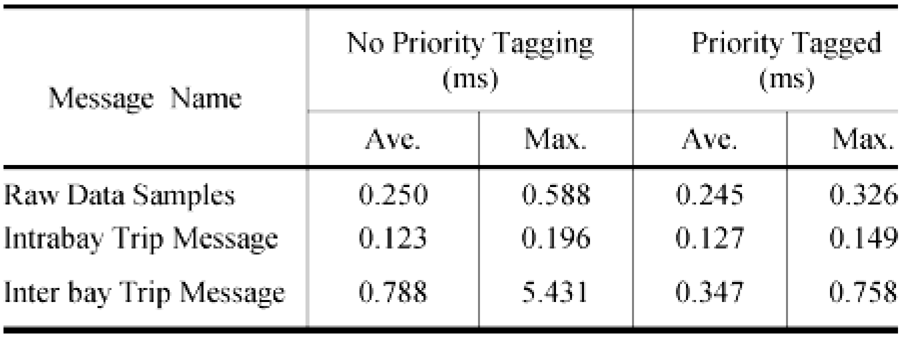
| Message Name | No Priority Tagging (ms) | | Priority Tagged (ms) | |
|---|---|---|---|---|
| | Ave. | Max. | Ave. | Max. |
| Raw Data Samples | 0.252 | 0.776 | 0.242 | 0.371 |
| Intrabay Trip Message | 0.127 | 0.196 | 0.120 | 0.150 |
| Interbay Trip Message | 0.550 | 1.729 | 0.262 | 0.545 |

It can be observed that Raw data sample and intra- bay trip messages have almost similar characteristics with the first topology at 10 Mbps. However, the maximum inter-bay trip message delay reaches 1.729 milliseconds without priority tagging. In this case it is worth doing a simulation to show network performance if another five feeder bay units are added to the above transformer switch in a network without changing any other configuration.

Table 4.7 illustrates the substation message delay of the 11 feeder-bay networks. This time the inter-bay trip message ETE delay reached 5.431 milliseconds under file transfer conditions and without priority tagging. However there is a significant time difference under priority tagging where the trip message ETE delay is reduced to 0.758 milliseconds. It is therefore recommended that for 10 Mbps networks the priority tagging must be included for the performance improvement to time for critical messages under heavy network traffic [5].

University of Cape

**Table 4.7 Expanded Economic Configuration – messages ETE delay [10]**



| Message Name | No Priority Tagging (ms) | | Priority Tagged (ms) | |
|---|---|---|---|---|
| | Ave. | Max. | Ave. | Max. |
| Raw Data Samples | 0.250 | 0.588 | 0.245 | 0.326 |
| Intrabay Trip Message | 0.123 | 0.196 | 0.127 | 0.149 |
| Inter bay Trip Message | 0.788 | 5.431 | 0.347 | 0.758 |

# 5. Laboratory Experiments

From the literature review, what has arisen is that GOOSE messages are assumed to be faster than the physical contact of the relays. The aim of the experiments is to prove or disprove this theory. Specifically, the experiments compare the speed of GOOSE messages with the speed of hardwired contacts of the relays, test a Breaker fail function using GOOSE messages instead of using the conventional hardwired contacts, and establish the interoperability of IEDs from different vendors using GOOSE message sharing.

## 5.1 Experiment 1: Comparison between GOOSE and physical contact outputs

The experiment consists of two F35 General Electric (GE) relays (relay-1 and relay-2), an Omicron test set, laptop and a Human-Machine Interface. All these devices communicate via an Ethernet switch. Since the Omicron test set is able to communicate with the GE relays using GOOSE messages, this proves that interoperability was met. Interoperability is the ability of IEDs from different suppliers or vendors to communicate or share information [24:8]. Figure 5.1 indicates the devices used in experiment -1.

Cape Town

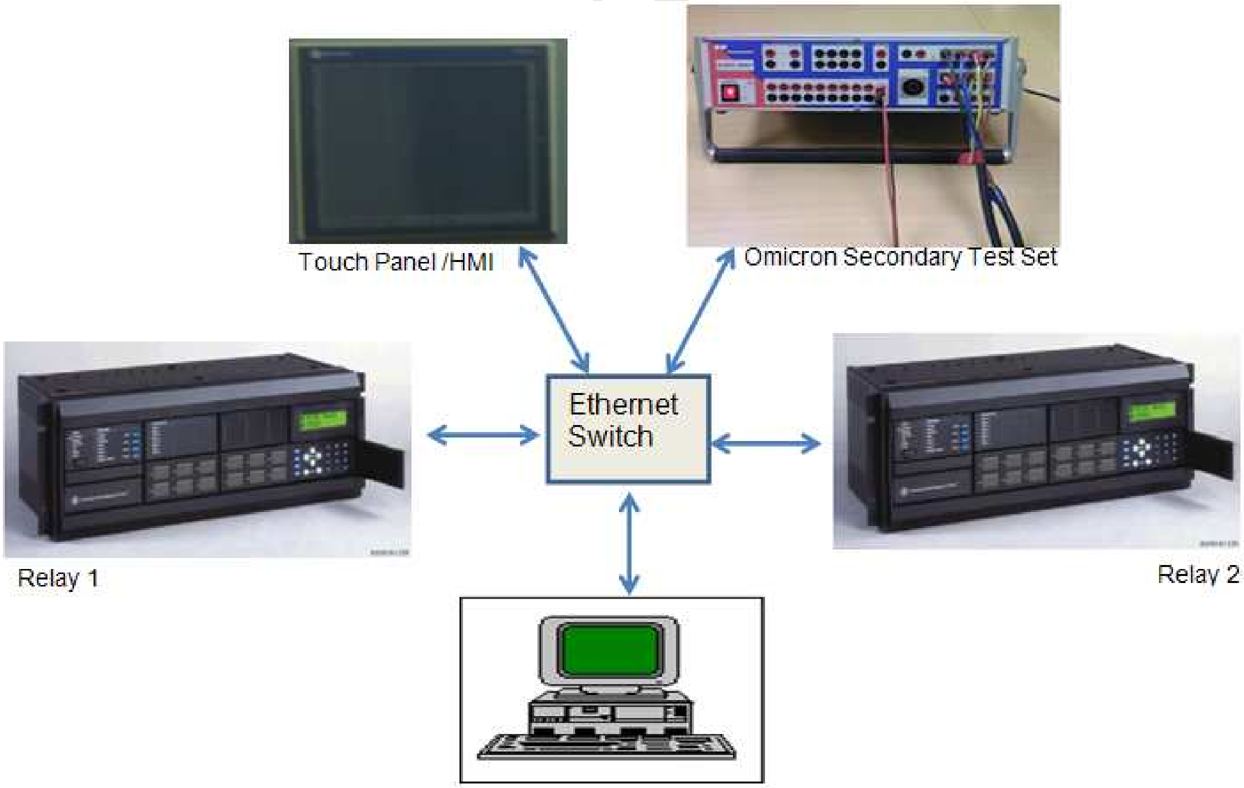


**Figure 5.1 The connection of devices in the Lab for the transmission of GOOSE and other messages.**

In experiment-1 the speed of GOOSE messages and physical contact outputs are compared. Figure 5.2 illustrates a setup for GOOSE versus hardwired contact output test.

In order for this to be achieved a point of reference or common starting point was created. The Omicron test set is used as a reference or common starting point for all the experiments that fall under experiment -1. To create a common starting point, an Omicron Binary output contact was closed resulting in a close on the F35 relay-1 contact input. From relay-1 contact input, the contact input is configured independently to relay-1 physical contact and GOOSE outputs.

Both F35 relay-1 physical contact output and GOOSE output are then used as inputs to the second relay called F35 relay-2. Therefore F35 relay-1 physical contact output and GOOSE output are configured to the F35 relay-2 physical contact input and GOOSE input respectively. The inputs of relay-2 are then configured to their contacts output; the relay-2 physical contact input is configured to its physical contact output, and relay-2 GOOSE input is configured to its GOOSE output.

The aim of experiment -1 is to compare which is faster between the GOOSE messages and physical contacts signals. To simplify the comparison, only the GOOSE and the physical contact outputs results of relay-2 were analysed in detail and not the results between the Omicron Binary output contact and relay-2 outputs. What is found to be important is that both the GOOSE and physical contacts have a common starting point which is the Omicron binary output. What happens in between is considered to be the contributing factors of the overall experiment.

Cape Town

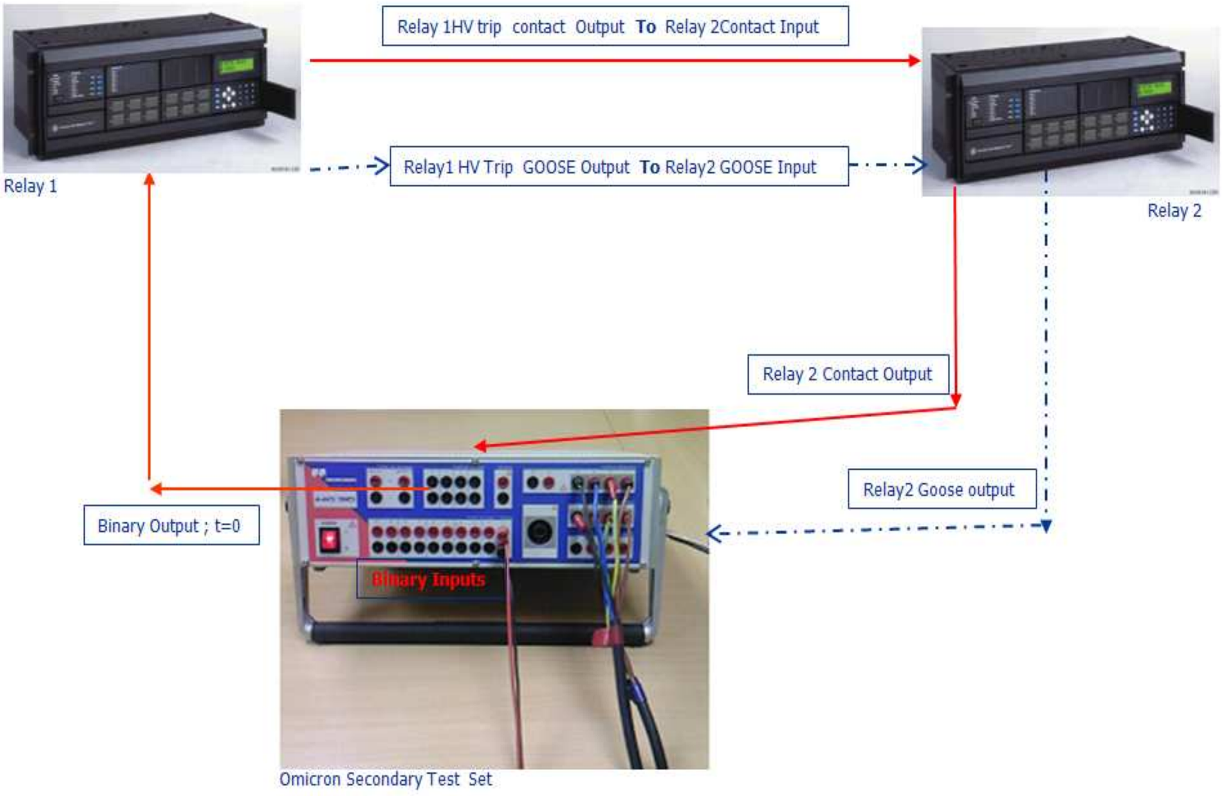


**Figure 5.2 Setup for GOOSE versus Hardwired contact output test**

### 5.1.1 Experiment 1.1a

For experiment 1.1a, contact inputs and outputs Form-C card is used on both GE relays: relay-1 and relay-2. The Form-C card of the GE relay contains the physical contact inputs and outputs and its contact output operation time is within 8 milliseconds. This card can easily be replaced or swapped with another one depending on the type of card required

The debounce time of contact inputs of both relays' cards are each set to 16 milliseconds. Denounce time determines the required time for the contact input to overcome the debouncing condition. The debouncing condition is when the contact is not stable in its status, e.g. when one tries to close the contact, it will first close and open several times, then after some milliseconds it will remain closed. This condition where the contact is fluctuating is called the debouncing condition. The debounce time setting is usually settable via software of the relay [25]. The normal settings on the GE relay is 16 milliseconds, which is the maximum setting value.

Now that the debounce and the operating times are defined; the experiment is as follows:

The experiment is started by closing the contact output of the Omicron test that is wired to the relay-1 physical contact input. The output of relay-1 is taken out as both physical contact output and the GOOSE output simultaneously, meaning that the output goes out in two forms, which are GOOSE and physical output.

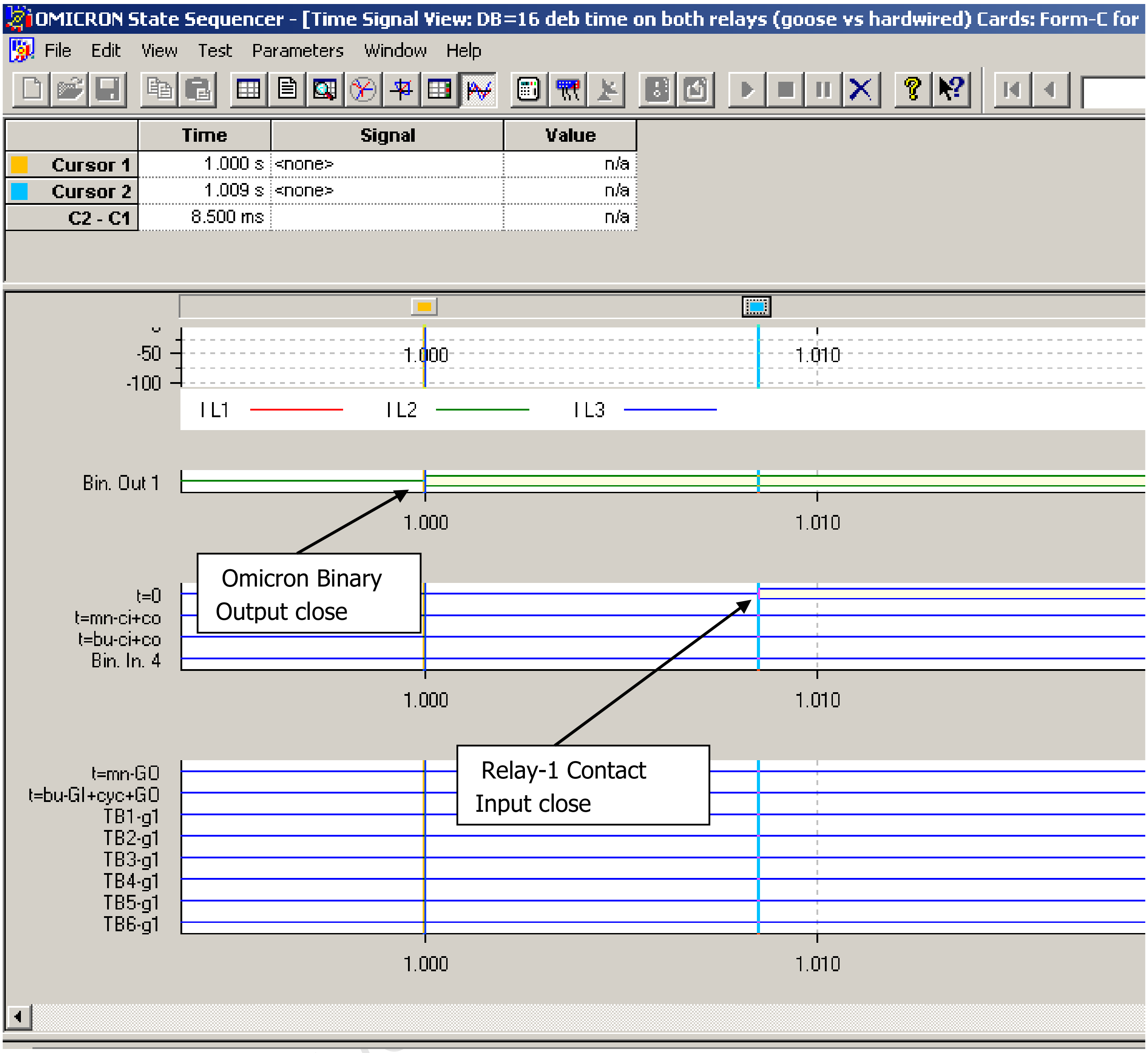
Both F35 relay-1 physical contact output and GOOSE output are then sent as inputs to relay-2. They are configured to the physical contact input and GOOSE input of relay-2. Since the inputs of relay-2 are configured to their contacts outputs; the outputs are picked up by the Omicron test set and the results for experiment 1.1a are therefore as follows:

A close on the Omicron binary output contact which results in the close on relay -1 contact input is shown in Figure 5.3. The close on contact input of relay-1 came after 8.5 milliseconds. This contact input is configured to both the GOOSE and physical contact output of relay-1, which are then independently configured to GOOSE and the physical contact input of relay-2. The GOOSE and physical contact input of relay-2 are then configured to the GOOSE and physical contact output of relay-2.

Figures 5.4 and 5.6 are shown but they will not be discussed or analysed in detail since the focus in this work is on Figure 5.3 (Omicron binary output close) and Figure 5.7 (time difference between relay-2 physical contact and GOOSE outputs).

Note that all the figures in this chapter consist of the following labels: IL1, IL2 and IL3. These labels represent red phase current, white phase current and blue phase current, respectively**.**

University of Cape Town



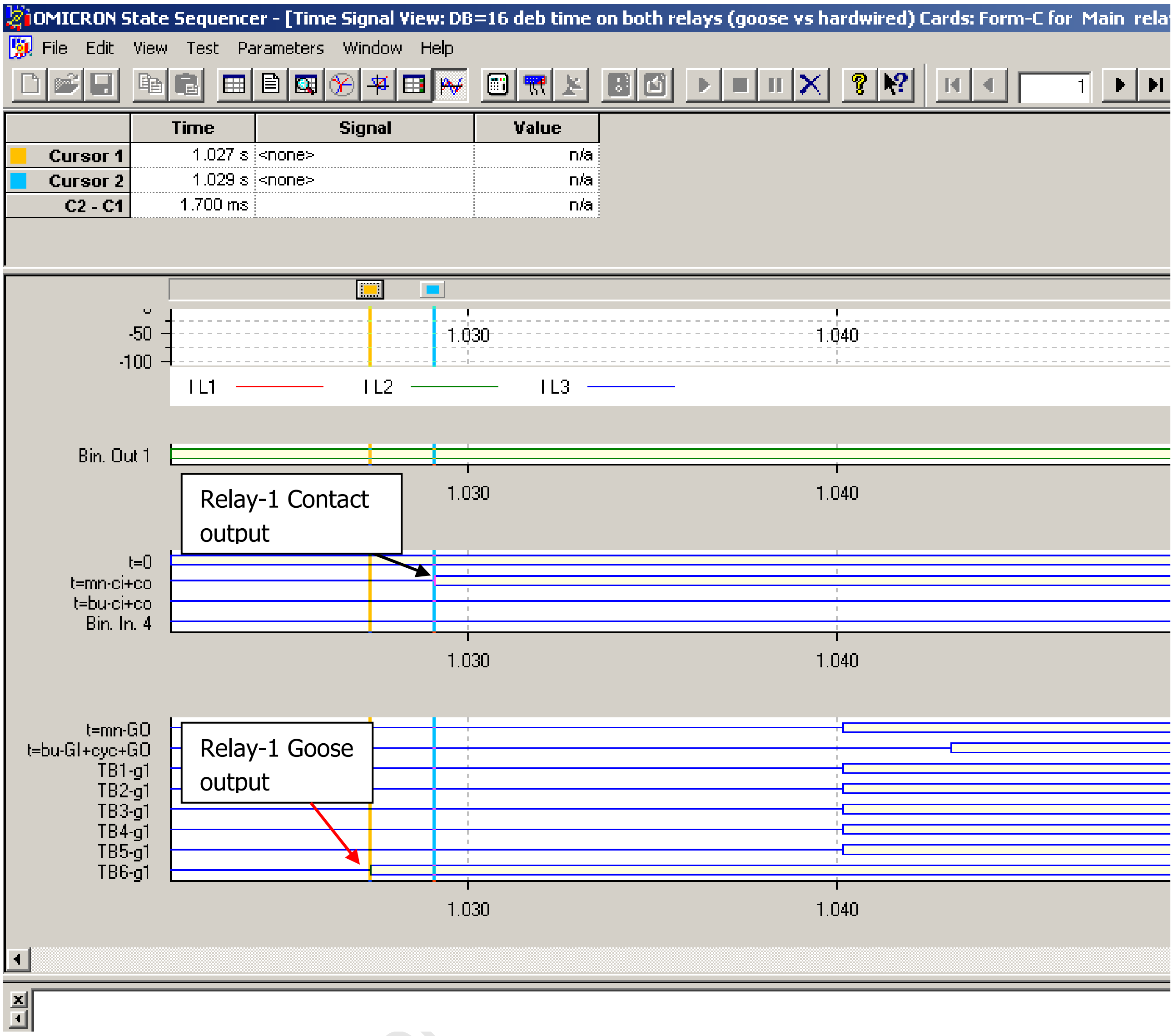
| | Time | Signal | Value |
|---|---|---|---|
| 🟧 Cursor 1 | 1.000 s | <none> | n/a |
| 🟦 Cursor 2 | 1.009 s | <none> | n/a |
| C2 - C1 | 8.500 ms | | n/a |

**Figure 5.3 Time difference between the Omicron Binary Output contact and relay-1 physical contact input close**

Figure 5.3 illustrates the time difference between the Omicron Binary Output contact and relay-1 physical contact input close.

The following labels from Figure 5.3 and the figures that follow are described as follows:

| | |
|---|---|
| “Bin. Out 1” | Omicron Binary Output Close |
| “t=0” | relay-1 time at which the relay-1 physical input contact closed |
| “t=mn-ci+co” | time at which relay-1 physical contact output is asserted |
| “t=bn-ci+co” | time at which relay-2 physical contact output is asserted |
| “t=mn-GO” | time at which relay-1 GOOSE output is asserted |
| “t=bu-GI+cyc+GO” | time at which relay-2 GOOSE output is asserted |
| “TB6-g1” | time at which relay-1 GOOSE contact output is asserted |



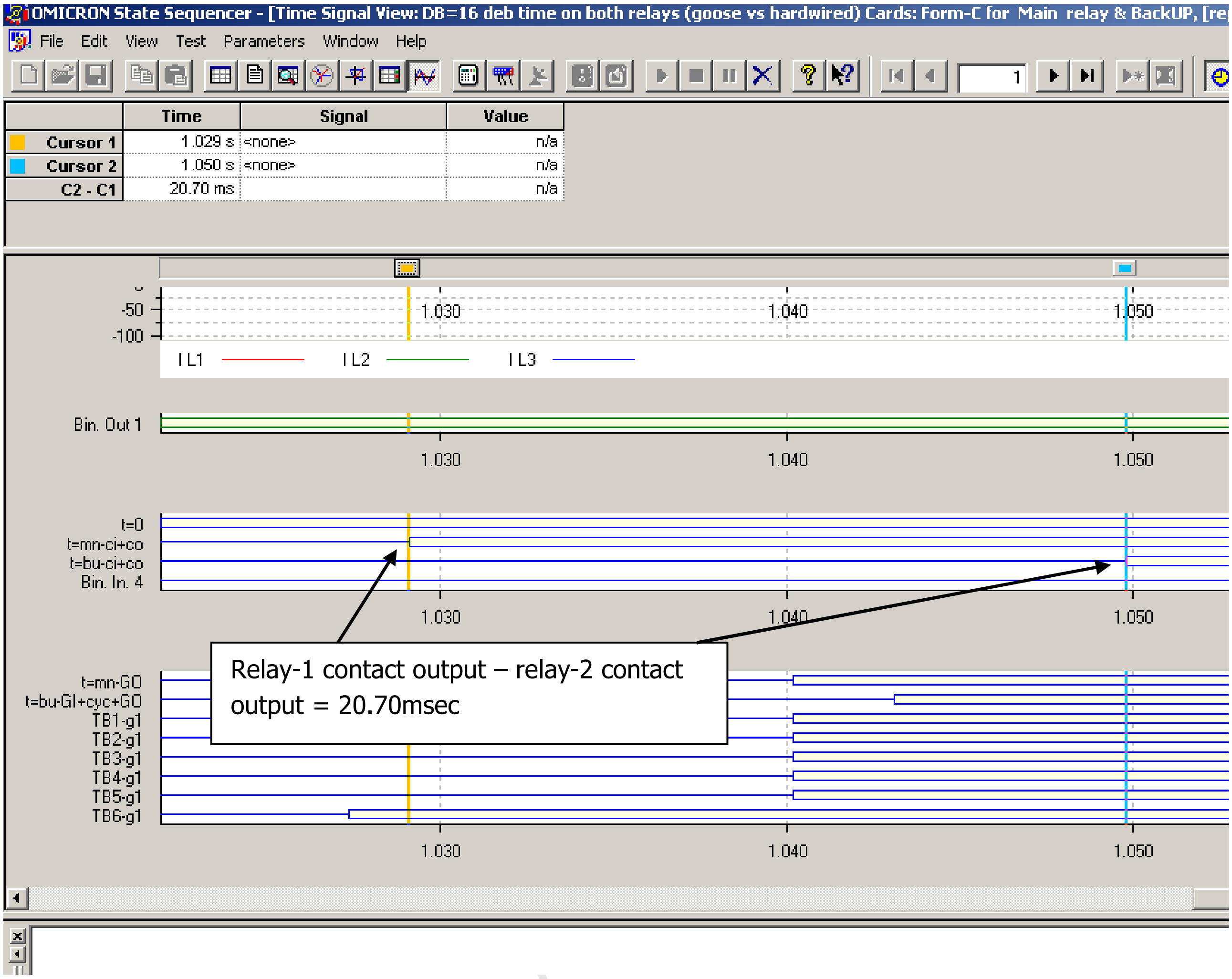
OMICRON State Sequencer - [Time Signal View: DB=16 deb time on both relays (goose vs hardwired) Cards: Form-C for  Main  rela

| | Time | Signal | Value |
|---|---|---|---|
| 🟧 Cursor 1 | 1.027 s | <none> | n/a |
| 🟦 Cursor 2 | 1.029 s | <none> | n/a |
| C2 - C1 | 1.700 ms | | n/a |

Figure 5.4 Time difference between the relay-1 physical contact and GOOSE output

Figure 5.4 illustrates that relay-1 GOOSE output was triggered faster than the physical output contact. GOOSE labelled "TB6- g1" which is a Trip GOOSE (yellow line) is faster than the relay-1 physical contact output (blue line) labelled "t = mn-ci+co" by 1.7 milliseconds.

Figure 5.5 illustrates the time difference between relay-1 physical contact output (yellow line) and relay-2 physical contact output (blue line).



| | Time | Signal | Value |
|---|---|---|---|
| 🟧 Cursor 1 | 1.029 s | <none> | n/a |
| 🟦 Cursor 2 | 1.050 s | <none> | n/a |
| C2 - C1 | 20.70 ms | | n/a |

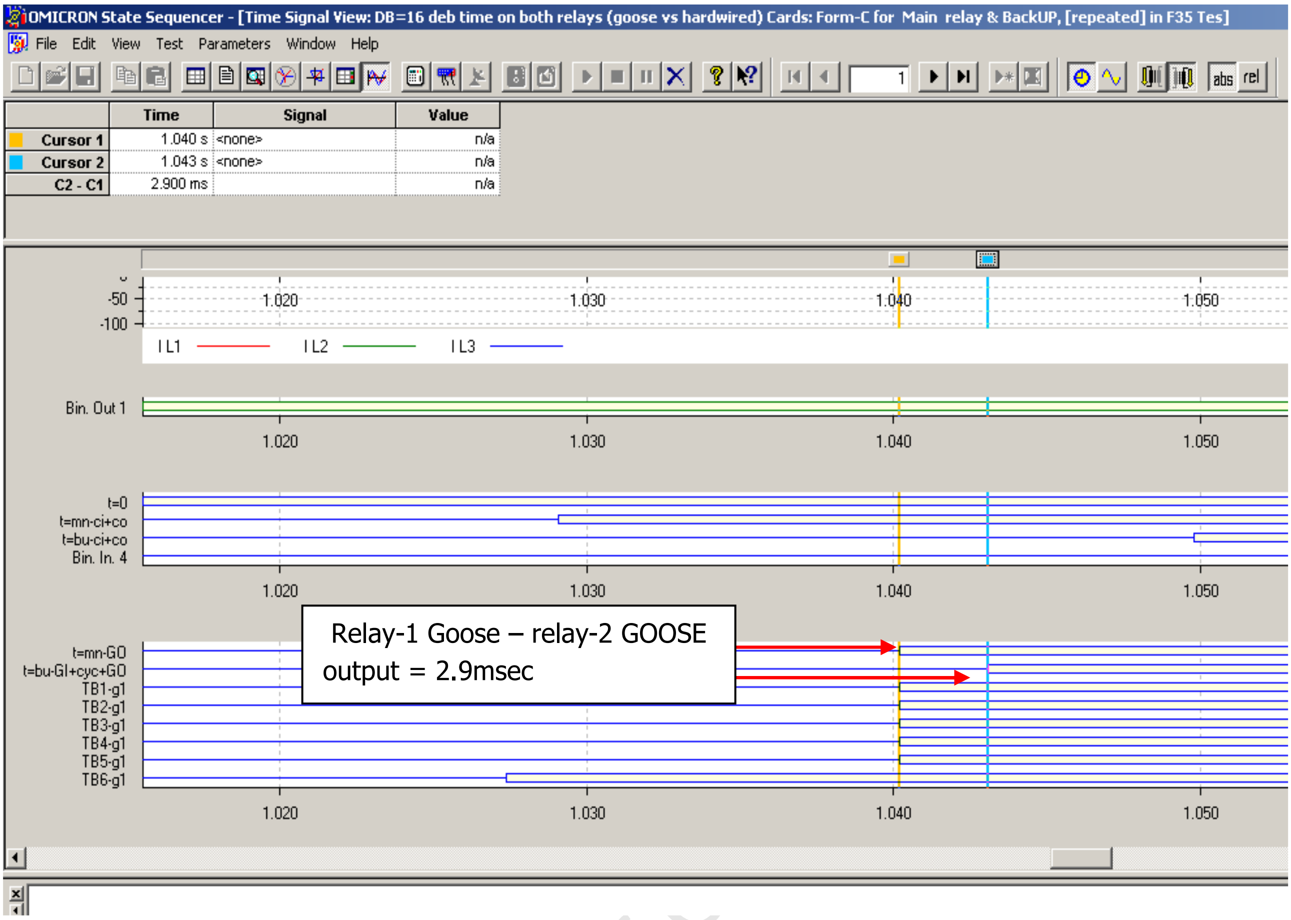
Relay-1 contact output – relay-2 contact output = 20.70msec

**Figure 5.5 Time difference between relay-1 physical contact output and relay-2 physical contact output**

From Figure 5.5, it can be observed that the time difference between relay-1 physical contact output and relay-2 physical contact output is 20.70 milliseconds. This time difference is about the expected time of 20 milliseconds since the relay-2 contact input debounce time is set to 16 milliseconds and the physical contact output operates within 4 milliseconds.

Universi

Figure 5.6 illustrates the time difference between relay-1 (yellow line) and relay-2 GOOSE (blue line) outputs.



OMICRON State Sequencer - [Time Signal View: DB=16 deb time on both relays (goose vs hardwired) Cards: Form-C for Main relay & BackUP, [repeated] in F35 Tes]

| | Time | Signal | Value |
|---|---|---|---|
| Cursor 1 | 1.040 s | <none> | n/a |
| Cursor 2 | 1.043 s | <none> | n/a |
| C2 - C1 | 2.900 ms | | n/a |

Relay-1 Goose – relay-2 GOOSE output = 2.9msec

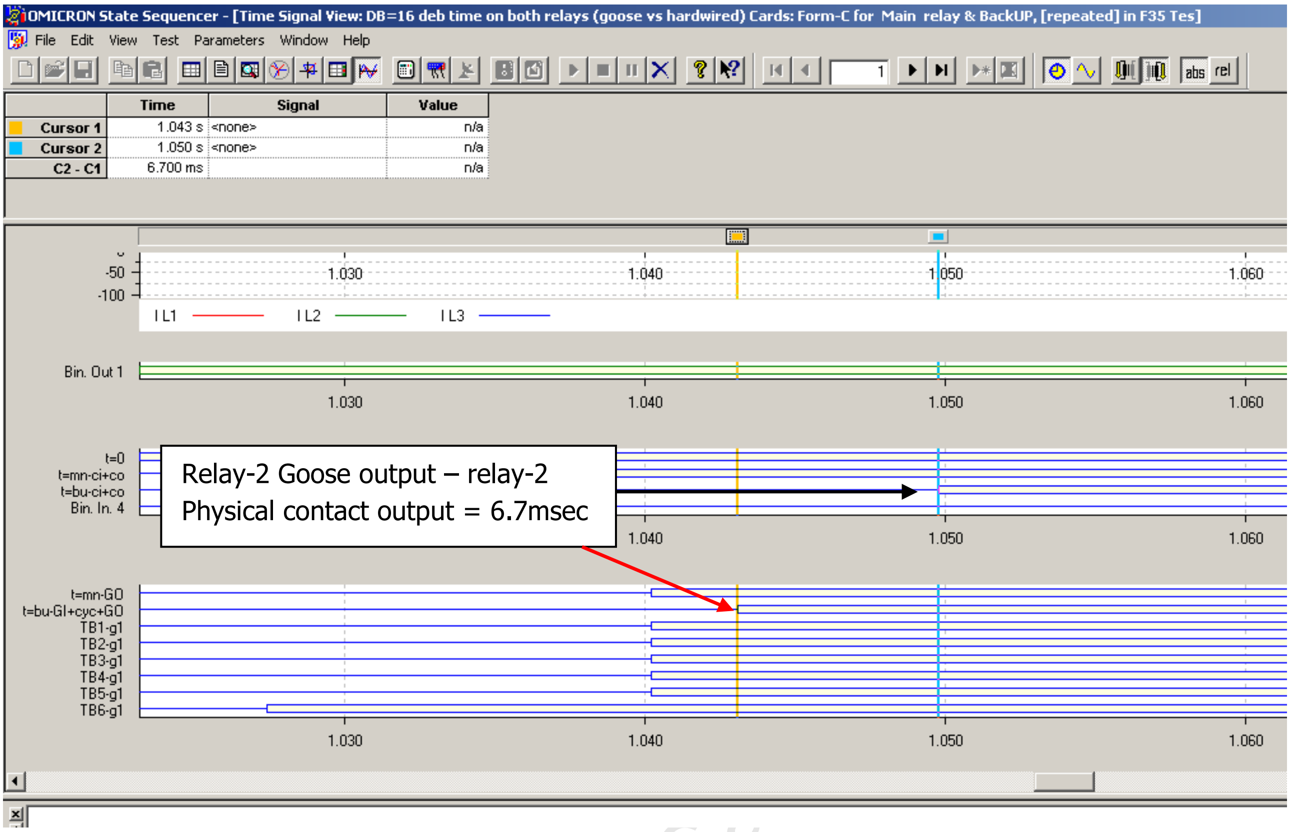
**Figure 5.6 Time difference between the relay-1 and relay-2 GOOSE outputs**

From Figure 5.6, it can be observed that the time difference between the relay-1 and relay-2 GOOSE outputs is 2.9 milliseconds.

Figure 5.7 illustrates the time difference between relay-2 physical contact (yellow line) and GOOSE (colour) outputs.

University of

.



|  | Time | Signal | Value |
|---|---|---|---|
| 🟧 Cursor 1 | 1.043 s | <none> | n/a |
| 🟦 Cursor 2 | 1.050 s | <none> | n/a |
| C2 - C1 | 6.700 ms |  | n/a |

Relay-2 Goose output – relay-2
Physical contact output = 6.7msec

**Figure 5.7 Time difference between the relay-2 physical contact output and GOOSE output**

According to Figure 5.7, it can be observed that the relay-2 GOOSE output (yellow line) is faster than its physical contact output (blue line) by 6.7 milliseconds.

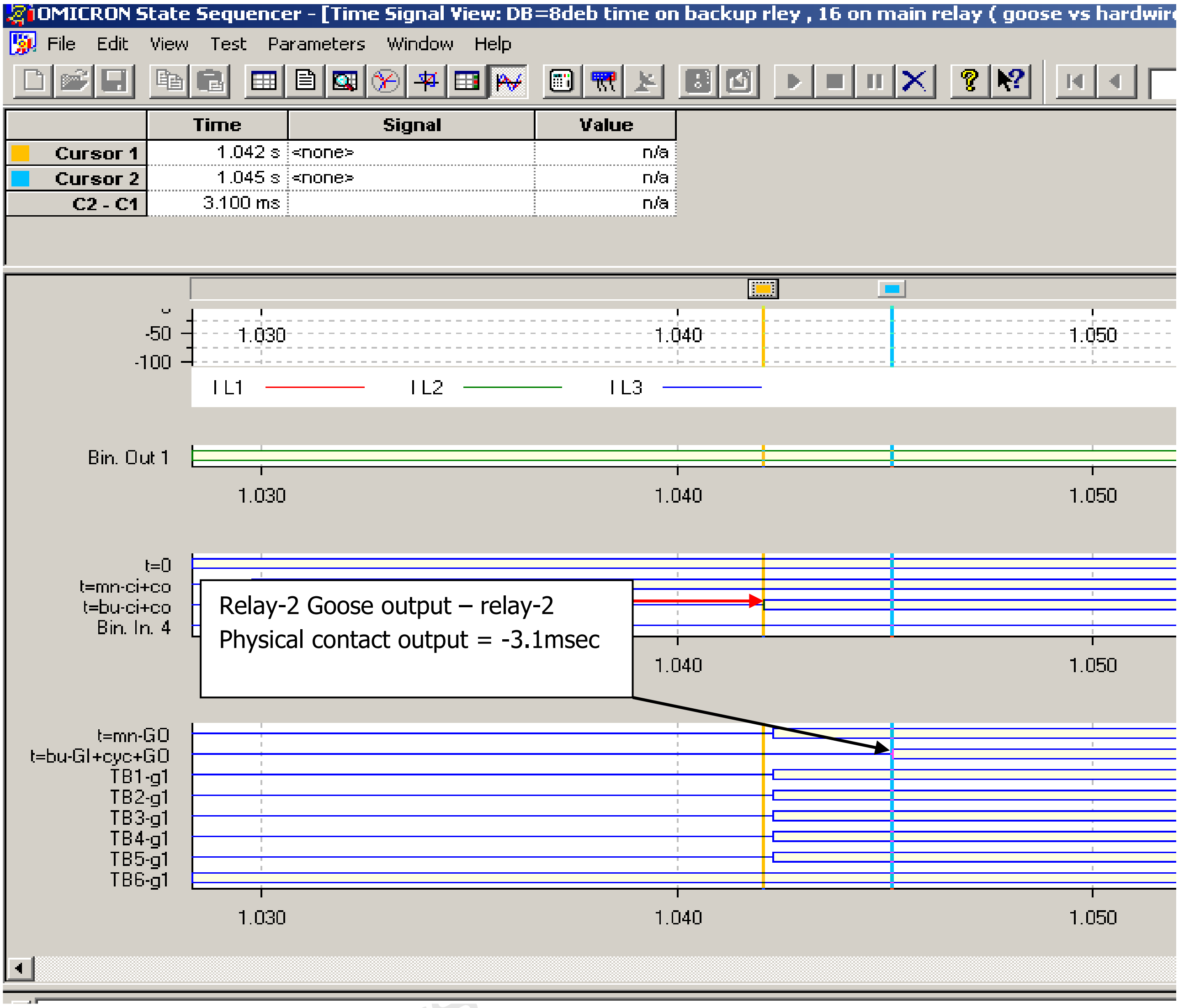
## 5.1.2 Experiment 1.1b

For experiment 1.1b, similar to experiment 1.1a; relay-1 and relay-2 use contact inputs and outputs Form-C card. The debounce time of contact inputs of relay-1 remains 16 milliseconds while the one for Relay-2 is decreased from 16 to 4 milliseconds. Everything else remains the same (i.e., Form-C contact output for both relays operating time being within 8 milliseconds).

Experiment 1.1b is run the same way as the experiment 1.1a.

Results for experiment 1.1b are shown in Figure 5.8

University of C



| | Time | Signal | Value |
|---|---|---|---|
| 🟨 Cursor 1 | 1.042 s | <none> | n/a |
| 🟦 Cursor 2 | 1.045 s | <none> | n/a |
| C2 - C1 | 3.100 ms | | n/a |

Relay-2 Goose output – relay-2
Physical contact output = -3.1msec

Figure 5.8 Time difference between the relay-2's contact output and GOOSE output

Univers

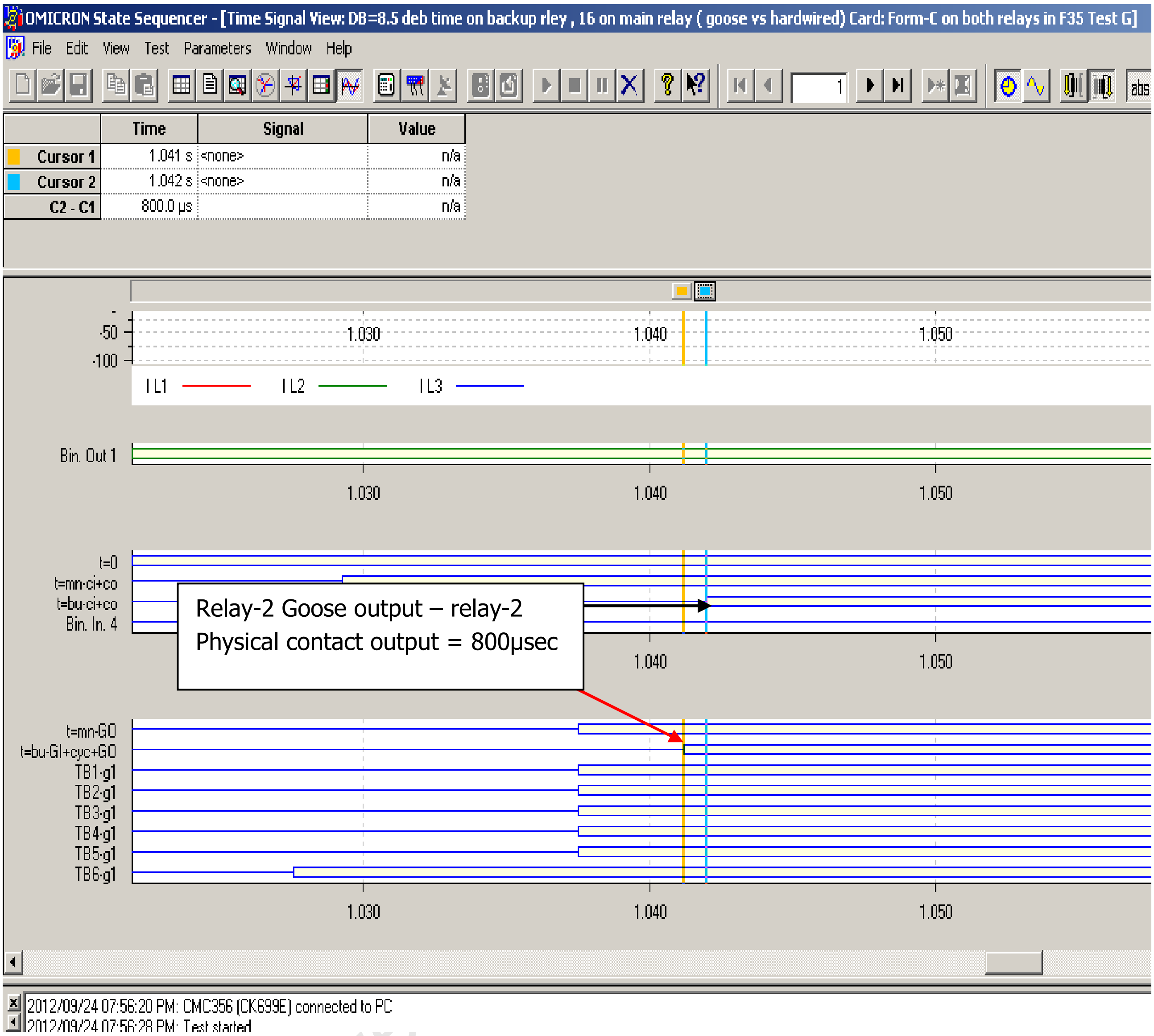
According to Figure 5.8, the relay-2's GOOSE output (blue line) is found to be slower than the physical output (yellow line) by (-) 3.1 milliseconds.

### 5.1.3 Experiment 1.1c

For experiment 1.1c, similar to experiment 1.1a, relay-1 and relay-2 use the contact inputs and outputs Form-C card. The debounce time of contact inputs of relay-1 remains 16 milliseconds while the one for relay-2 is increased to 4.5 from 4 milliseconds. Everything else remains the same as in experiment 1.1a.

.

Experiment 1.1c is run the same way as the experiment 1.1a. The results for experiment 1.1c are shown in Figure 5.9:



| | Time | Signal | Value |
|---|---|---|---|
| 🟧 Cursor 1 | 1.041 s | <none> | n/a |
| 🟦 Cursor 2 | 1.042 s | <none> | n/a |
| C2 - C1 | 800.0 µs | | n/a |

Relay-2 Goose output – relay-2 Physical contact output = 800µsec

```
2012/09/24 07:56:20 PM: CMC356 (CK699E) connected to PC
2012/09/24 07:56:28 PM: Test started
```

**Figure 5.9 Time difference between the relay-2 contact output and GOOSE output**

From Figure 5.9, it can be observed that the relay-2 GOOSE output (yellow line) is faster than the physical contact output (blue line) by 800 microseconds.

The analysis of experiment 1.1a, 1.1b and 1.1c are in Section 5.2.

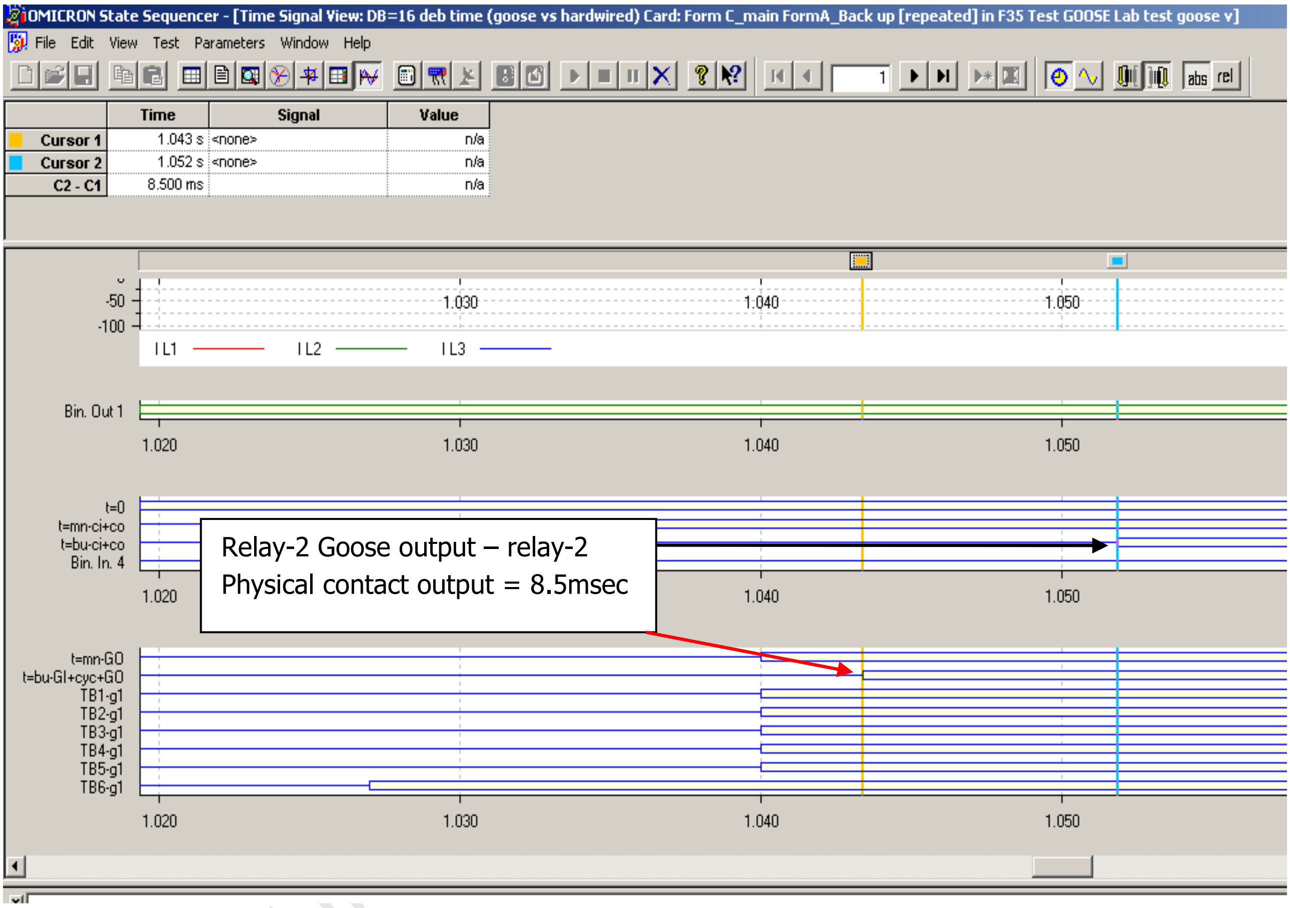
### 5.1.4 Experiment 1.2a

For experiment 1.2a, contact inputs and outputs Form-C card is used on GE relay-1, while contact inputs and outputs Form-A card is used for GE relay-2. The Form-A card of the GE relay contains the physical contact inputs and outputs and its contact output operation time is within 4 milliseconds and its contact input is settable via software. The Form-C cards contact output operation time is within 8 milliseconds. The debounce time of contact inputs of both the relay cards are set to 16 milliseconds each.

Now that the debounce and the operating times are defined; the experiment is as follows:

The experiment is started by closing the contact output of omicron test that is wired to the relay-1 physical contact input. The output of relay-1 goes out as both physical contact output and the GOOSE output simultaneously and independently.

Both F35 relay-1 physical contact output and GOOSE outputs are configured to the inputs of relay-2, which are physical contact input and GOOSE input of relay-2, respectively. The inputs of relay-2 are configured to its contacts outputs; these outputs of relay-2 are therefore asserted by relay-2 inputs. The results for experiment 1.2a are therefore as follows:



OMICRON State Sequencer - [Time Signal View: DB=16 deb time (goose vs hardwired) Card: Form C_main FormA_Back up [repeated] in F35 Test GOOSE Lab test goose v]

| | Time | Signal | Value |
|---|---|---|---|
| 🟧 Cursor 1 | 1.043 s | <none> | n/a |
| 🟦 Cursor 2 | 1.052 s | <none> | n/a |
| C2 - C1 | 8.500 ms | | n/a |

Relay-2 Goose output – relay-2 Physical contact output = 8.5msec

Figure 5.10 Time difference between the relay-2 contact output and GOOSE output

From Figure 5.10, it can be observed that the relay-2 GOOSE output (yellow line) is faster than the physical contact output (blue line) by 8.5 milliseconds.

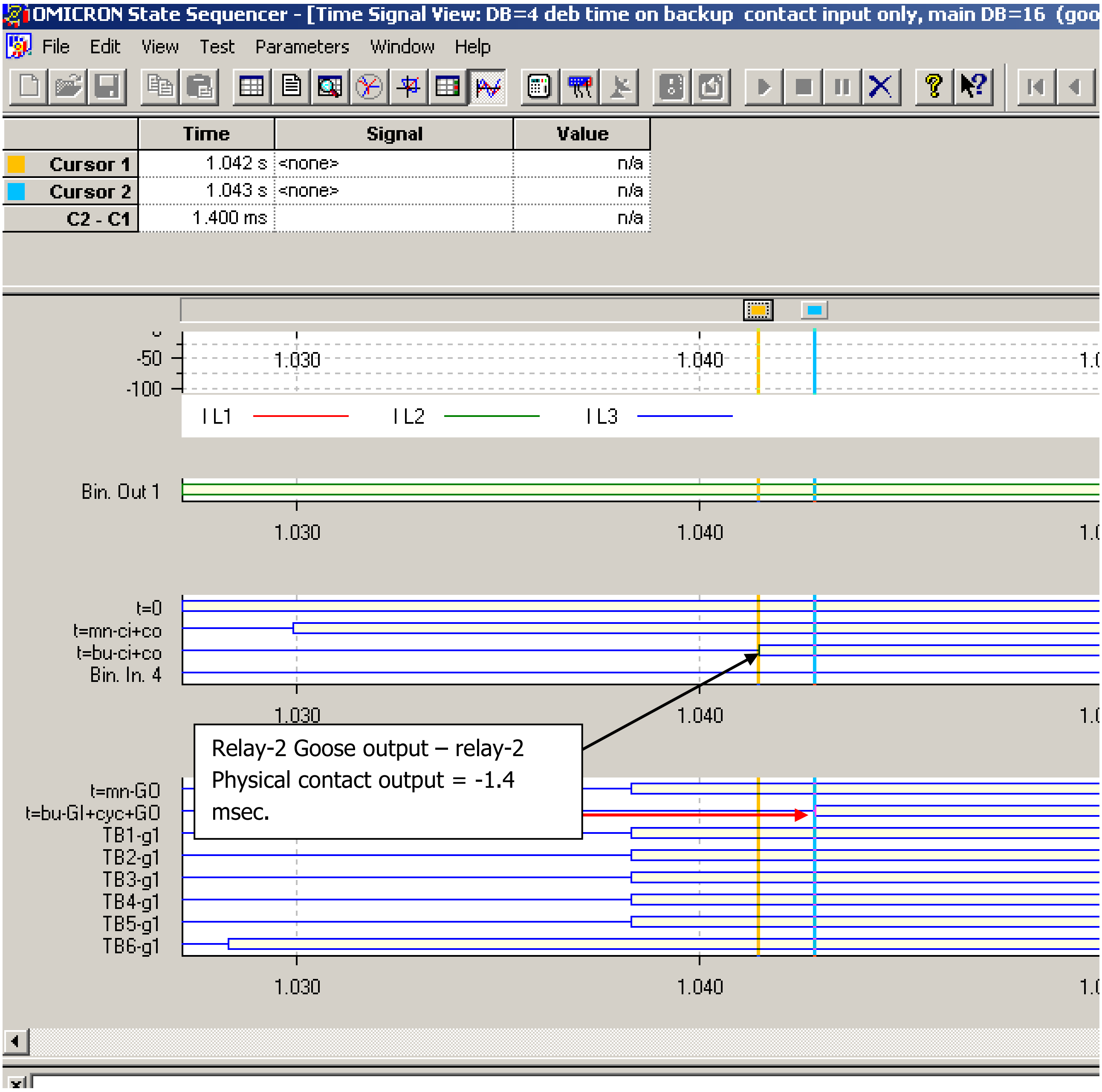
Refer to Appendix A for detailed results of experiment 1.2a.

## 5.1.5 Experiment 1.2b:

For experiment 1.2b, similar to experiment 1.2a; the GE relay-1 using contact inputs and outputs Form-C card and relay-2 are using contact inputs and outputs Form- A card. The Form-C card's contact output operation time is within 8 milliseconds, while the operating time of Form-A cards is within 4 milliseconds.

The debounce time of relay-1's contact input cards is set to 16 milliseconds while the debounce time of relay-2's contact input is set to 8 milliseconds.

Experiment 1.2b is run the same way as the experiment 1.2a and the results are as follows:


OMICRON State Sequencer - [Time Signal View: DB=4 deb time on backup  contact input only, main DB=16  (goo

| | Time | Signal | Value |
|---|---|---|---|
| 🟨 Cursor 1 | 1.042 s | <none> | n/a |
| 🟦 Cursor 2 | 1.043 s | <none> | n/a |
| C2 - C1 | 1.400 ms | | n/a |

Relay-2 Goose output – relay-2 Physical contact output = -1.4 msec.

Figure 5.11 Time difference between the relay-2's contact output and GOOSE output

In Figure 5.11, it is observed that relay-2 GOOSE output (blue line) is now slower than the physical contact output (yellow line) by (-) 1.4 milliseconds.

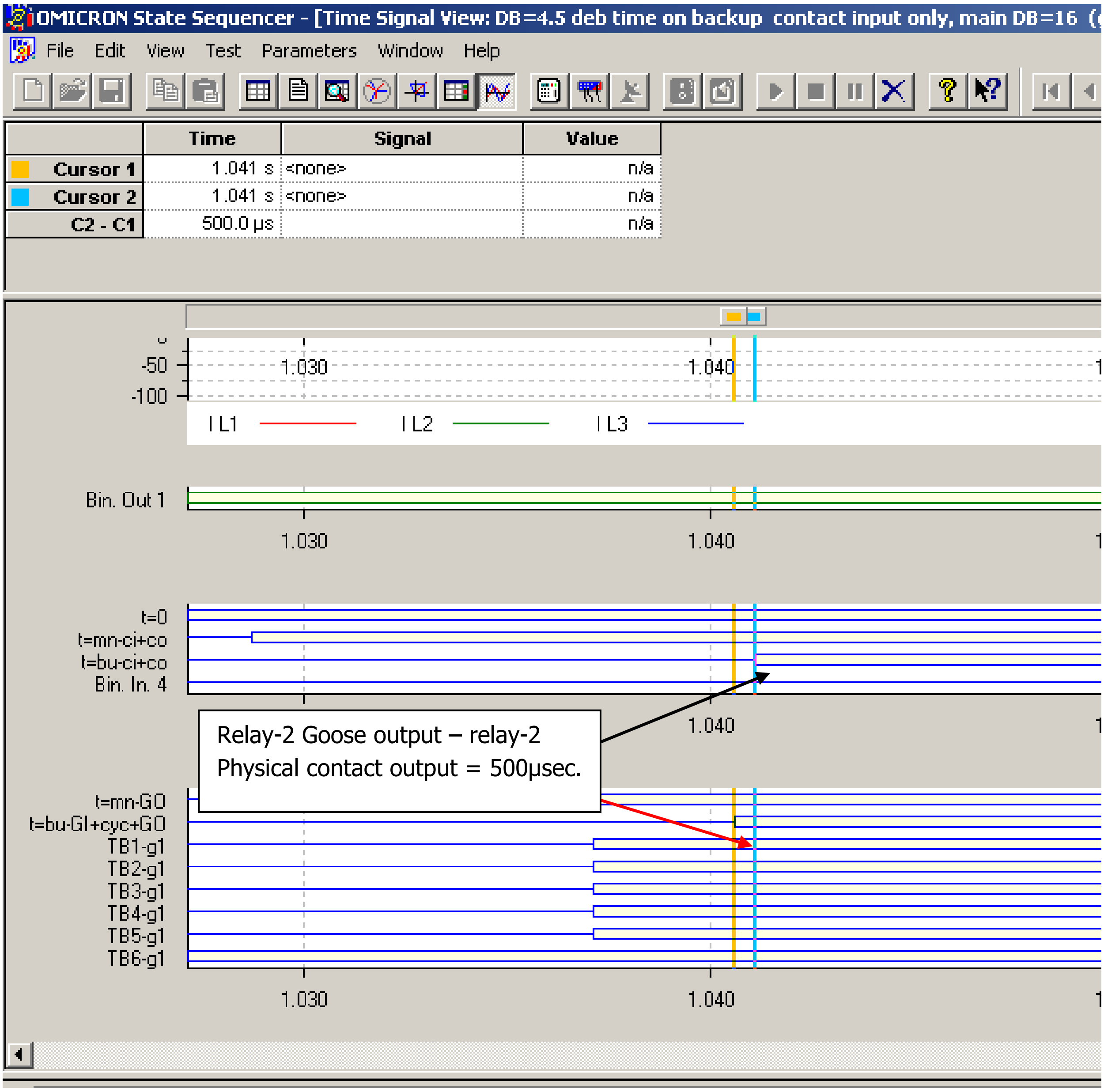
### 5.1.6 Experiment 1.2c

For experiment 1.2c, similar to experiment 1.2a; the relay-1 is using contact inputs and outputs Form-C card and relay-2 is using contact inputs and outputs Form A- card.

The Form-C card's contact output operation time is within 8 milliseconds, while the operating time of Form-A cards is within 4 milliseconds.

The debounce time of relay-1's contact input cards is set to 16 milliseconds while the debounce time of relay-2's contact input is set to 8.5 milliseconds.

Experiment 1.2c is run the same way as the experiment 1.2a. The results are as follows:



OMICRON State Sequencer - [Time Signal View: DB=4.5 deb time on backup  contact input only, main DB=16 (

| | Time | Signal | Value |
|---|---|---|---|
| 🟨 Cursor 1 | 1.041 s | <none> | n/a |
| 🟦 Cursor 2 | 1.041 s | <none> | n/a |
| C2 - C1 | 500.0 µs | | n/a |

Relay-2 Goose output – relay-2
Physical contact output = 500µsec.

**Figure 5.12 Time difference between the relay-2 contact output and GOOSE output**

Figure 5.12 indicates that relay-2 GOOSE output (yellow line) is faster than the physical contact output (blue line) by 500 microseconds.

The analysis of experiments 1.2a, 1.2b and 1.2c are in the results analysis section 5.2.

## 5.2 Analysis of the results of experiments 1

Table 5.1 is the summary of the results for experiments 1.1a to 1.1c.

**Table 5.1 Results of GOOSE versus Hardwired contact output test for experiments 1.1a to 1.1c**

| | Experiment 1.1a | Experiment 1.1b | Experiment 1.1c |
|---|---|---|---|
| relay-1 Contact Output Type | Form –C card , operates with in 8msec | Form –C card , operates with in 8msec | Form –C card , operates with in 8msec |
| relay-1 Contact Input Debounce Time | 16 msec | 16 msec | 16 msec |
| relay-2 Contact Output Type | Form –C card , operates with in 8msec | Form –C card , operates with in 8msec | Form –C card , operates with in 8msec |
| relay-2 Contact Output Debounce Time | 16 msec | 4msec | 4.5msec |
| relay-1 GOOSE output (Master Trip) | 1.027 sec | 1.028 sec | 1.028 sec |
| relay-1 GOOSE output (HV Breaker Trip) | 1.040 sec | 1.042 sec | 1.038 sec |
| relay-1 Contact Output | 1.029 sec | 1.030 sec | 1.029 sec |
| relay-2 GOOSE output | 1.043 sec | 1.045 sec | 1.041sec |
| relay-2 Contact Output | 1.050 sec | 1.042 sec | 1.042 sec |
| Time difference between relay-1 GOOSE (Master Trip) and Contact/Physical output | 1. 700 msec | 1.7 sec | 1.7 msec |
| Time difference between relay-1 and relay-2 relay Contact outputs | 20.70 msec | 12.30 msec | 12.70 msec |
| Time difference between relay-1 ( HV Trip) and relay-2 GOOSE | 2.9 msec | 2.9 msec | 3.7 msec |
| Time difference between relay-2 GOOSE and Contac/Physical output | 6.7 msec | -3.1msec | 800µsec |

From Table 5.1, it can be observed that with relay-2 using a Form–C card and a debounce time of 16 milliseconds as illustrated in experiment 1.1a ( 4th row) , the results of relay-2 GOOSE output is faster than that of the physical contact output by 6.7 milliseconds (last row). When the debounce time is reduced from 16 to 4 milliseconds as illustrated in experiment 1.1b, the GOOSE output becomes slower than physical contact output by 3.1 milliseconds; this is when the combination of the operating time and the debounce time adds up to 12 milliseconds in total. When Operating Time is equal to 8 milliseconds (3rd row line) and debounce time is equal to 4 milliseconds (4th row)

When the debounce time is again increased to 4.5 milliseconds as illustrated in experiment 1.1c, the GOOSE output becomes faster again than physical contact output, by 800 microseconds. From the results of experiment 1.1b and 1.1c one can observe a border between the 4 milliseconds and 4.5 milliseconds.

Experiments 1.2a, 1.2b and 1.2c were also performed and are indicated in Table 5.2.

Town

**Table 5.2 Results of GOOSE versus hardwired contact output test for Experiments 1.2a to 1.2c**

| | Experiment 1.2a | Experiment 1.2b | Experiment 1.2c |
|---|---|---|---|
| relay-1 Contact Output Type | Form –C card , operates with in 8msec | Form–C card , operates with in 8msec | Form –C card , operates with in 8msec |
| relay-1 Contact Input Debounce Time | 16 msec | 16 msec | 16 msec |
| relay-2 Contact Output Type | Form –A card , operates with in 4msec | Form –A card , operates with in 4msec | Form –A card , operates with in 4msec |
| relay-2 Contact Output Debounce Time | 16msec | 8msec | 8.5msec |
| relay-1 GOOSE output (Master Trip) | 1.027 sec | 1.028 sec | 1.027 sec |
| relay-1 GOOSE output (HV Breaker Trip) | 1.040 sec | 1.038 sec | 1.037 sec |
| relay-1 Contact Output | 1.029 sec | 1.030 sec | 1.029 sec |
| relay-2 | 1.043 sec | 1.043 sec | 1.041 sec |
| GOOSE output | 1.052 sec | 1.042 sec | 1.041 sec |
| relay-2 Relay/Relay2 - | 1.6 msec | 1.6 msec | 1.7 msec |
| Contact Output | 23.3 msec | 11.6 msec | 12.40msec |
| Time difference between relay-1 GOOSE (Master Trip) and Contact/Physical output | 3.4 msec | 4.6 msec | 3.5 msec |
| Time difference between relay-1 and relay-2 relay Contact outputs | 8.5 msec | -1.4 msec | 500 µsec |

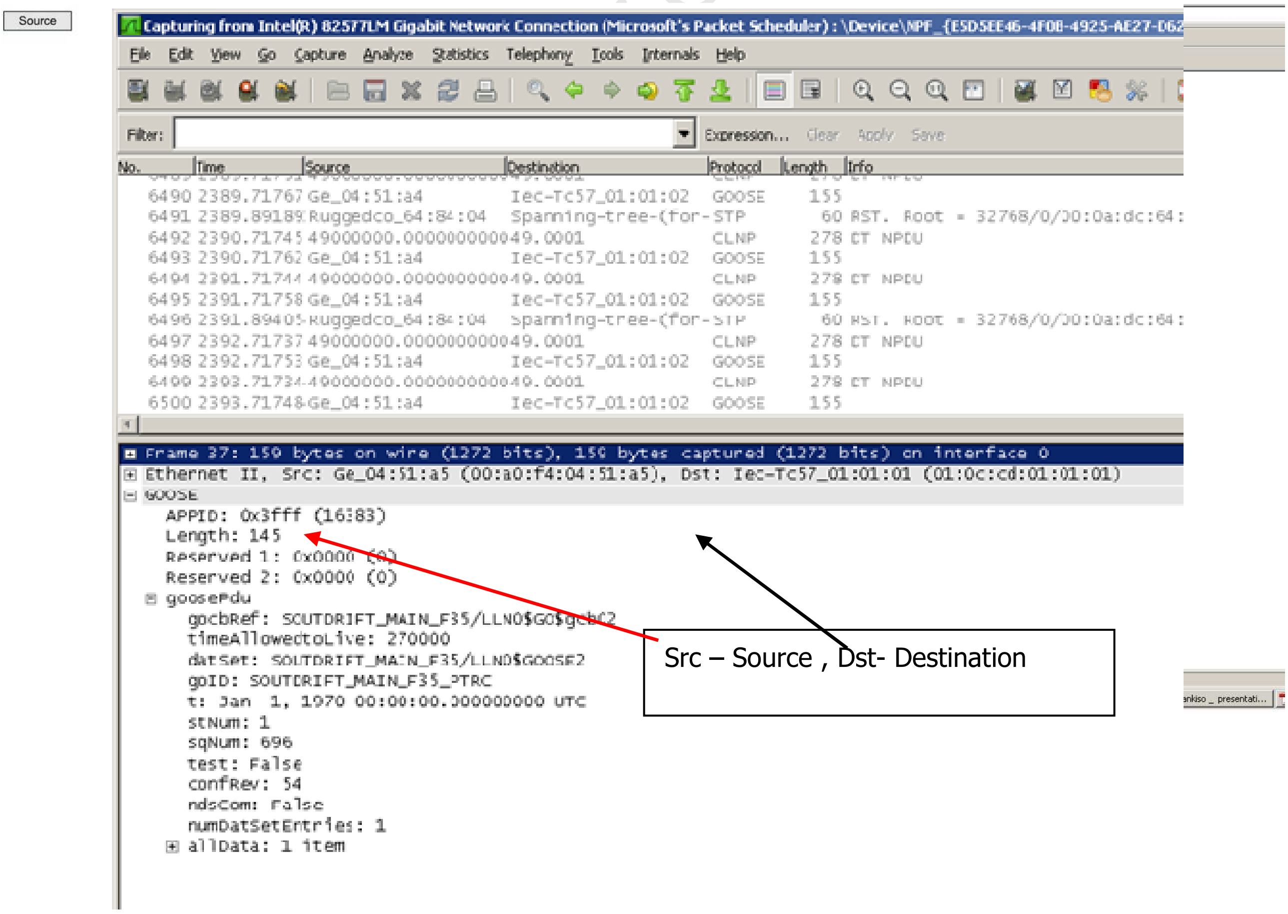
From Table 5.2, it can be observed that with relay-2 using a Form –A card and a debounce time of 16 milliseconds as illustrated in experiment 1.2a, the GOOSE output is faster than the physical contact output by 8.5 milliseconds (last row in the table).

When the debounce time is reduced from 16 (experiment 1.2a 4th row) to 8 (experiment 1.2b 4th row) milliseconds as illustrated in experiment 1.2b, GOOSE output is slower than the physical contact output by 1.4 milliseconds.
In experiment 1.2c, the debounce time in relay-2 is now increased from 8 milliseconds to 8.5 milliseconds. The GOOSE output becomes faster than physical contact output by 500 microseconds. From these results of experiment 1.2b and 1.2c one can observe a border between the GOOSE being faster or slower than the physical contact output.

## 5.3 Experiment 2: Comparison between GOOSE and physical contact outputs with sampled values traffic introduced in a network

Experiment 2 is a repeat of experiment 1 but with an introduction of sampled values in a network. Sampled values are analogue values such as currents and voltages that are sampled. The main purpose of introducing sampled values is to introduce more traffic in a network, thus determining the effect of samples values traffic on the performance of GOOSE messages. Figure 5.13 below illustrates a network without sampled values traffic. The software that is used to monitor the traffic is wire-shark.

Cape Town



```
6490 2389.71767 Ge_04:51:a4                Iec-Tc57_01:01:02  GOOSE   155
6491 2389.89189 Ruggedco_64:84:04          Spanning-tree-(for-STP      60 RST. Root = 32768/0/00:0a:dc:64:
6492 2390.71745 49000000.000000000049.0001                    CLNP    278 CT NPDU
6493 2390.71762 Ge_04:51:a4                Iec-Tc57_01:01:02  GOOSE   155
6494 2391.71744 49000000.000000000049.0001                    CLNP    278 CT NPDU
6495 2391.71758 Ge_04:51:a4                Iec-Tc57_01:01:02  GOOSE   155
6496 2391.89405 Ruggedco_64:84:04          Spanning-tree-(for-STP      60 RST. Root = 32768/0/00:0a:dc:64:
6497 2392.71737 49000000.000000000049.0001                    CLNP    278 CT NPDU
6498 2392.71753 Ge_04:51:a4                Iec-Tc57_01:01:02  GOOSE   155
6499 2393.71734 49000000.000000000049.0001                    CLNP    278 CT NPDU
6500 2393.71748 Ge_04:51:a4                Iec-Tc57_01:01:02  GOOSE   155
```

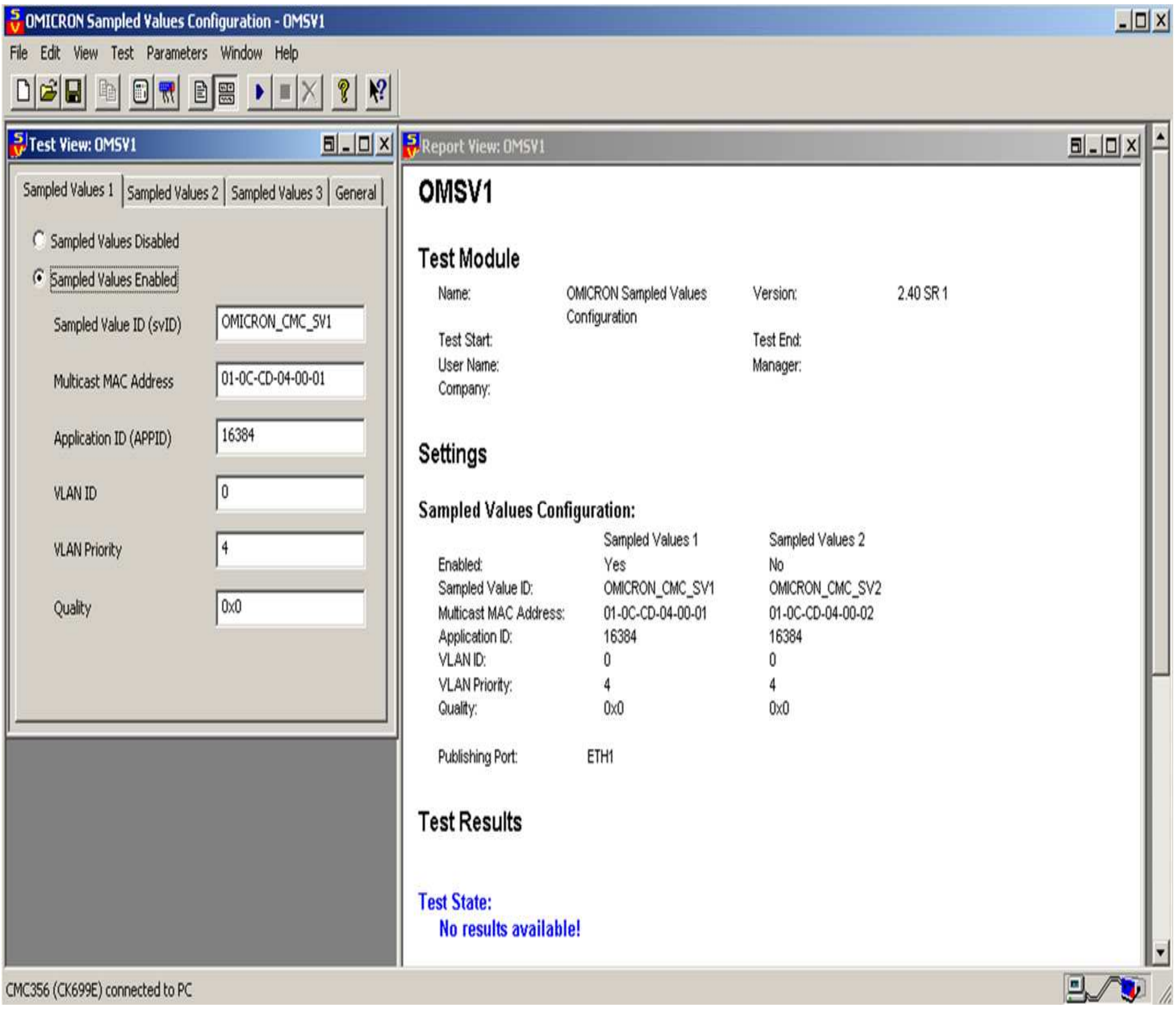
```
⊞ Frame 37: 159 bytes on wire (1272 bits), 159 bytes captured (1272 bits) on interface 0
⊞ Ethernet II, Src: Ge_04:51:a5 (00:a0:f4:04:51:a5), Dst: Iec-Tc57_01:01:01 (01:0c:cd:01:01:01)
⊟ GOOSE
    APPID: 0x3fff (16383)
    Length: 145
    Reserved 1: 0x0000 (0)
    Reserved 2: 0x0000 (0)
  ⊟ goosePdu
      gocbRef: SOUTDRIFT_MAIN_F35/LLN0$GO$gcbC2
      timeAllowedtoLive: 270000
      datSet: SOUTDRIFT_MAIN_F35/LLN0$GOOSE2
      goID: SOUTDRIFT_MAIN_F35_PTRC
      t: Jan  1, 1970 00:00:00.000000000 UTC
      stNum: 1
      sqNum: 696
      test: False
      confRev: 54
      ndsCom: False
      numDatSetEntries: 1
    ⊞ allData: 1 item
```

Src – Source , Dst- Destination

**Figure 5.13 The traffic before introducing sampled values**

Figure 5.13 shows that the goose messages originating from Omicron test set (labelled " src: Ge_04:51:a5 (00:a0:fa:04:51:a5). 00:a0:fa:04:51:a5 is the MAC or the physical address of Omicron test set . Src stands for source and Dst stands for destination. The Src MAC address identifies where the message is originated or sent from and Dst MAC address where the message is sent to. Dst : IEC–Tc57_ 01:01:01 (01:0c:cd:01:01:01) is the destination MAC address which is the F35 relay in this instance.

The sampled values are configured in the Omiron Test Universe software under " Omicron Samples Values Configuration". Figure 5.14 illustates how the samples values are configured in the Omicron Sampled Values Configuration file. The sampled values file name is "Omicron_CMC_SV1".



○ Sampled Values Disabled
● Sampled Values Enabled

Sampled Value ID (svID): OMICRON_CMC_SV1
Multicast MAC Address: 01-0C-CD-04-00-01
Application ID (APPID): 16384
VLAN ID: 0
VLAN Priority: 4
Quality: 0x0

**OMSV1**

**Test Module**

Name: OMICRON Sampled Values Configuration
Version: 2.40 SR 1
Test Start:
Test End:
User Name:
Manager:
Company:

**Settings**

**Sampled Values Configuration:**

| | Sampled Values 1 | Sampled Values 2 |
|---|---|---|
| Enabled: | Yes | No |
| Sampled Value ID: | OMICRON_CMC_SV1 | OMICRON_CMC_SV2 |
| Multicast MAC Address: | 01-0C-CD-04-00-01 | 01-0C-CD-04-00-02 |
| Application ID: | 16384 | 16384 |
| VLAN ID: | 0 | 0 |
| VLAN Priority: | 4 | 4 |
| Quality: | 0x0 | 0x0 |

Publishing Port: ETH1

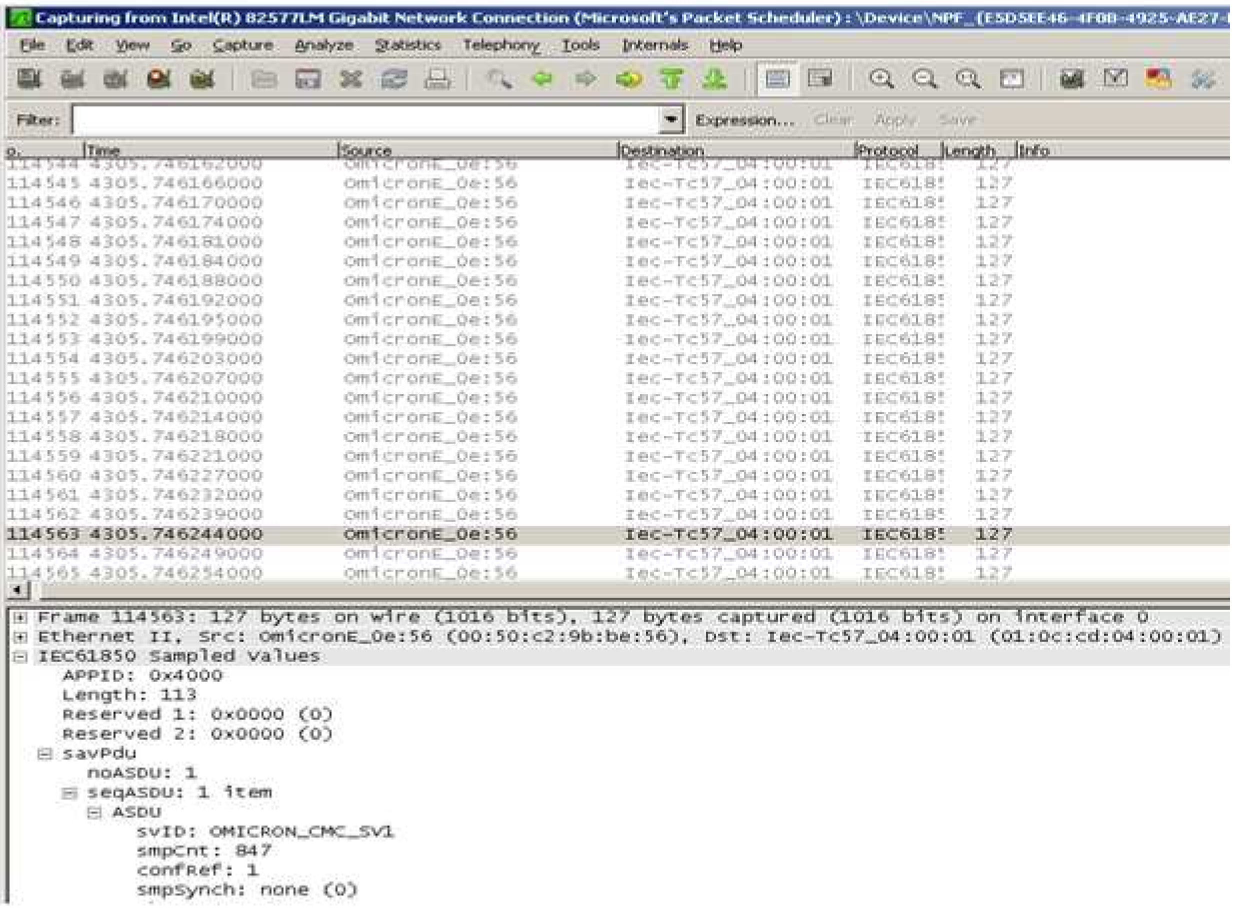
**Test Results**

Test State:
No results available!

CMC356 (CK699E) connected to PC

**Figure 5.14 Omicro sampled values configuration file**

Figure 5.15 illustates the network with sampled values traffic.



```
o.     Time               Source             Destination             Protocol  Length  Info
114544 4305.746162000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114545 4305.746166000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114546 4305.746170000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114547 4305.746174000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114548 4305.746181000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114549 4305.746184000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114550 4305.746188000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114551 4305.746192000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114552 4305.746195000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114553 4305.746199000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114554 4305.746203000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114555 4305.746207000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114556 4305.746210000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114557 4305.746214000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114558 4305.746218000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114559 4305.746221000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114560 4305.746227000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114561 4305.746232000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114562 4305.746239000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114563 4305.746244000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114564 4305.746249000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
114565 4305.746254000     OmicronE_0e:56     Iec-Tc57_04:00:01       IEC618!   127
```

```
⊞ Frame 114563: 127 bytes on wire (1016 bits), 127 bytes captured (1016 bits) on interface 0
⊞ Ethernet II, Src: OmicronE_0e:56 (00:50:c2:9b:be:56), Dst: Iec-Tc57_04:00:01 (01:0c:cd:04:00:01)
⊟ IEC61850 Sampled Values
    APPID: 0x4000
    Length: 113
    Reserved 1: 0x0000 (0)
    Reserved 2: 0x0000 (0)
  ⊟ savPdu
      noASDU: 1
    ⊟ seqASDU: 1 item
      ⊟ ASDU
          svID: OMICRON_CMC_SV1
          smpCnt: 847
          confRef: 1
          smpSynch: none (0)
```

Figure 5.15 The network with Omicron sampled values traffic

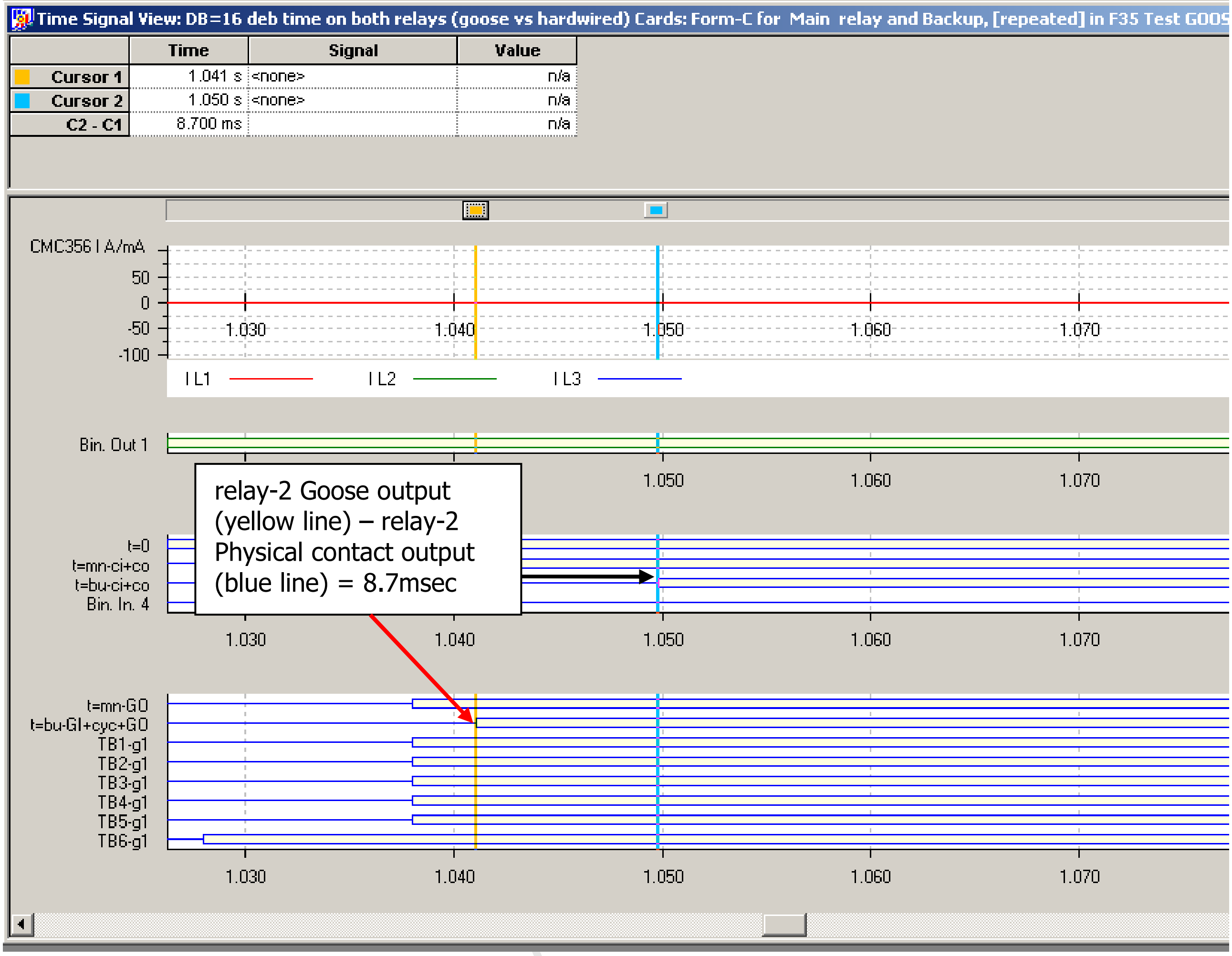
University o

Figure 5.15 illustrates that the network is now filled with Omicron sampled values and the MAC address 01-50-C2-9b-be-56 is from Omicron device and the destination is still the F35 relay with the Mac address 01:0c:cd:01:01:01.

## 5.3.1 Experiment 2.1

Experiment 2.1 is a repeat of experiment 1.1a but with the Omicron sampled values in the network. The results will therefore be compared to the results in experiments 1.1a to 1.1c to check the influence of the sampled values traffic on the performance of GOOSE messages.

Figure 5.16 illustrates the results of the time between the relay-2 GOOSE and hardwired contact after the sampled values traffic is introduced in the network.



Time Signal View: DB=16 deb time on both relays (goose vs hardwired) Cards: Form-C for Main relay and Backup, [repeated] in F35 Test GOOS

| | Time | Signal | Value |
|---|---|---|---|
| 🟨 Cursor 1 | 1.041 s | <none> | n/a |
| 🟦 Cursor 2 | 1.050 s | <none> | n/a |
| C2 - C1 | 8.700 ms | | n/a |

relay-2 Goose output (yellow line) – relay-2 Physical contact output (blue line) = 8.7msec

**Figure 5.16 The time between the relay-2 goose and hardwired contact with the sampled values traffic introduced in the network.**

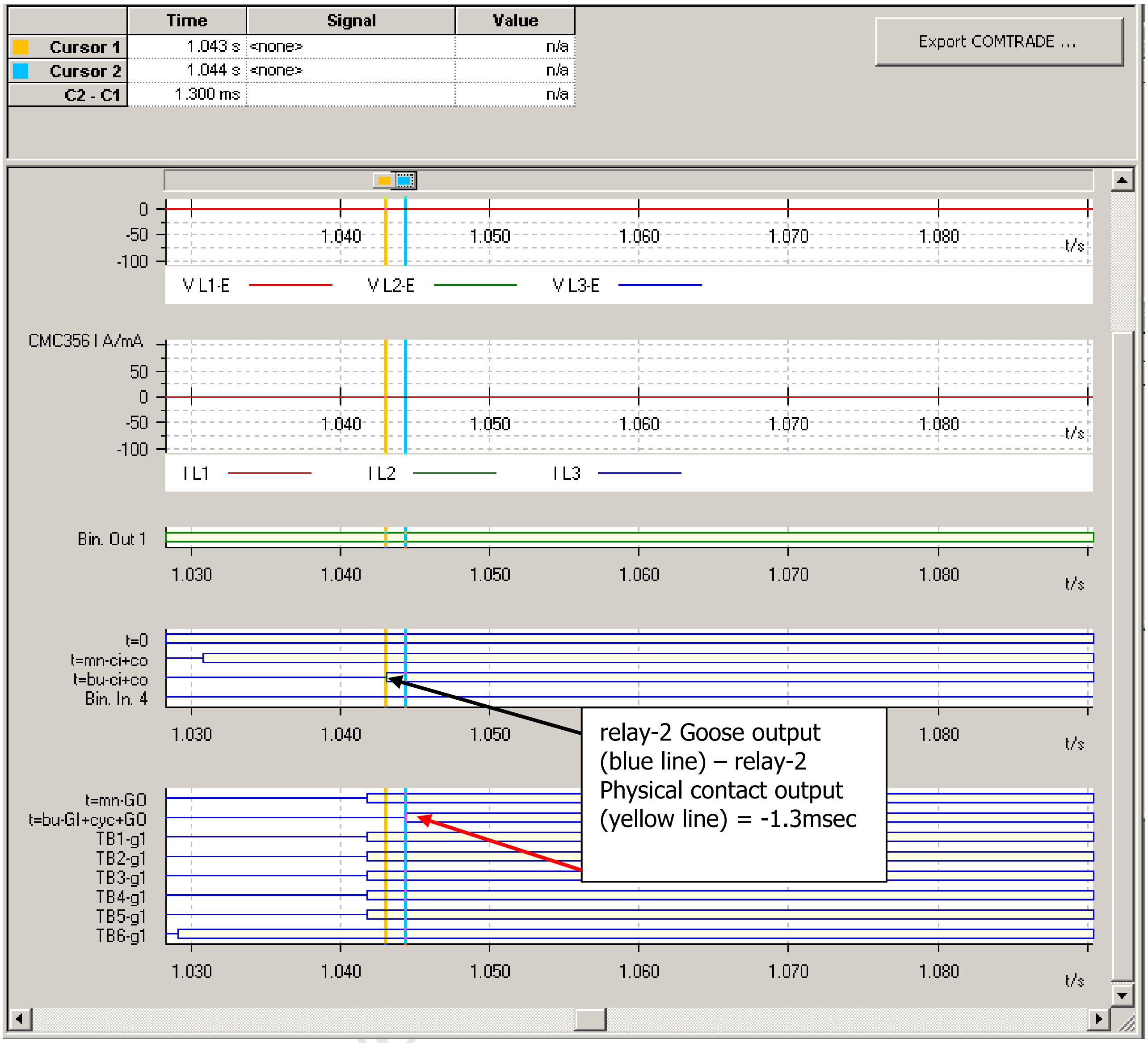
University

When comparing Figure 5.7 (no sampled values traffic) and Figure 5.16 (with sampled values traffic), it can be observed that the relay-2 relay GOOSE output is faster than the physical contact output in both cases. In Figure 5.7 the time difference is 6.7 milliseconds and in Figure 5.16 the time difference is 8.7 milliseconds. The introduction of the sampled values traffic has increased the GOOSE output time. This is in a case where both relays' debounce time is set to 16 milliseconds and the operating time of both the relays contact output is 8 milliseconds since they contain the Form-C contact inputs and outputs cards.

## 5.3.2 Experiment 2.2

Experiment 2.2 is the same as experiment 1.1b except that the sampled values traffic is now introduced in the network. The debounce time of relay-2 is reduced from 16 to 4 milliseconds while that of relay-1 remains at 16 milliseconds.

Figure 5.17 illustrates the results of the time between the relay-2 GOOSE and hardwired contact after sampled values traffic is introduced in the network.



|  | Time | Signal | Value |
|---|---|---|---|
| 🟨 Cursor 1 | 1.043 s | <none> | n/a |
| 🟦 Cursor 2 | 1.044 s | <none> | n/a |
| C2 - C1 | 1.300 ms |  | n/a |

relay-2 Goose output (blue line) – relay-2 Physical contact output (yellow line) = -1.3msec

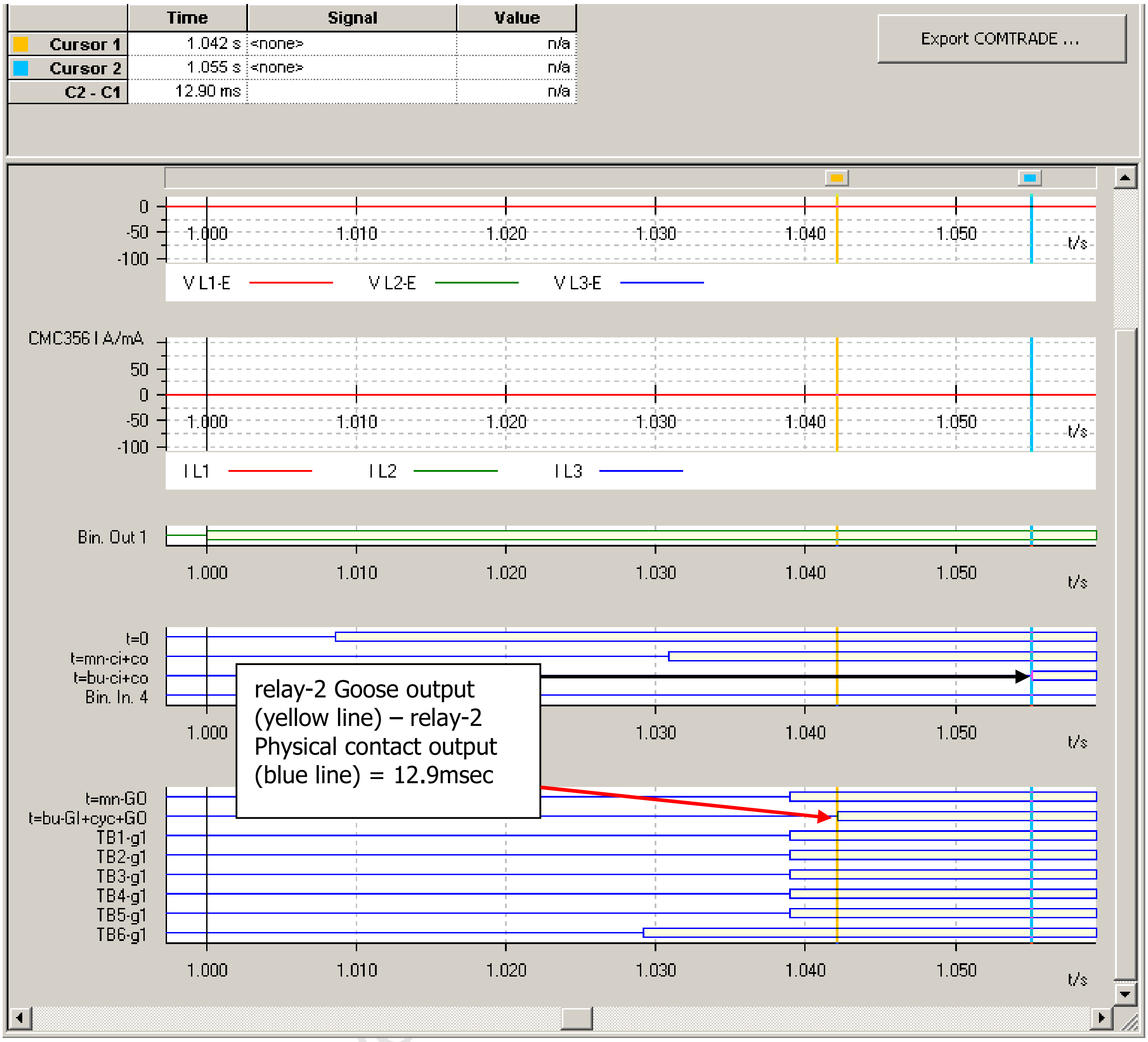
**Figure 5.17 The time between the relay-2 goose and relay-2 hardwired contact for experiment 2.2.**

When comparing Figure 5.8 and Figure 5.17 (where there is sampled values traffic) it can be observed that the relay-2 GOOSE in Figure 5.17 output is slower than the physical contact output as in Figure 5.8. In Figure 5.8 the time difference is (-) 3.1 milliseconds and in Figure 5.17 the time difference is (-) 1.3 milliseconds.

### 5.3.3 Experiment 2.3

Experiment 2.3 is the same as experiment 1.1c, except that the sampled values traffic is now introduced in the network. The debounce time of relay-2 is increased from 4 to 4.5 milliseconds, while that of relay-1 remains at 16 milliseconds.

Figure 5.18 illustrates the results of the time between the relay-2 GOOSE and hardwired contact after the sampled values traffic is introduced in the network.



| | Time | Signal | Value |
|---|---|---|---|
| 🟨 Cursor 1 | 1.042 s | <none> | n/a |
| 🟦 Cursor 2 | 1.055 s | <none> | n/a |
| C2 - C1 | 12.90 ms | | n/a |

relay-2 Goose output (yellow line) – relay-2 Physical contact output (blue line) = 12.9msec

**Figure 5.18 The time between the relay-2 goose and hardwired contact for experiement 2.3.**

When comparing Figure 5.9 and Figure 5.18 (where there is sampled values traffic), it can be observed that in Figure 5.18 the relay-2 GOOSE output is still faster than the physical contact output as in Figure 5.9. However, in Figure 5.9 the time difference is 800 microseconds and in Figure 5.18 the time difference is 12.90 milliseconds.

## 5.4 Analysis of the results of experiment 2

According to the three tests that are simulated in section 5.3, the introduction of sampled values traffic affects the results of the relay-2 GOOSE output versus physical contact output. The impact is not very significant. This is because in each test, if the GOOSE output was faster than the physical contact output before the introduction of the sampled values traffic, the GOOSE is still faster when the sampled values traffic is introduced. Again if GOOSE output was slower before the sampled values traffic was introduced, the GOOSE output is still slower when the sampled values traffic is introduced.

## 5.5 Experiment 3: Circuit breaker fail operation

The aim of this experiment is to demonstrate the breaker fail functionality using GOOSE messages instead of using the conventional method. In a conventional scheme, the breaker fail functionality operates when the circuit breaker (CB) that is supposed to clear the fault, fails to do so. The output contact of the breaker fail that is wired to the bus zone panel (bus zone panel is the panel where all the protections relays/IEDs connected to the same busbar are all wired) will then close, ensuring that an input to the bus zone panel is high. These will result in the bus zone panel tripping all the CBs connected to it. The signal moves from the relay of the feeder that experienced the fault to the bus zone panel, then the bus zone panel will send a trip signal to all protection relays/IEDs connected to the bus zone.

The operation of breaker fail function in this experiment is different in that there is no interface such as a bus zone panel. In this experiment, a GOOSE message that is carrying breaker fail signal was sent from the relay of the feeder that is experiencing a breaker fail, to the relay that is subscribed to this relay.

This experiment has two relays, relay-1 and relay-2. Relay-1 is the relay that will experience breaker fail from its breaker and relay-2 is the relay that was sent a GOOSE command to trip its CB due to the breaker fail experienced in relay-1. This is called bus strip operation. If there were more than one relay subscribed to relay-1, all these other relays will also receive the breaker fail signal and will therefore trip. See Figure 5.19 for the setup for the circuit breaker fail operation experiment.

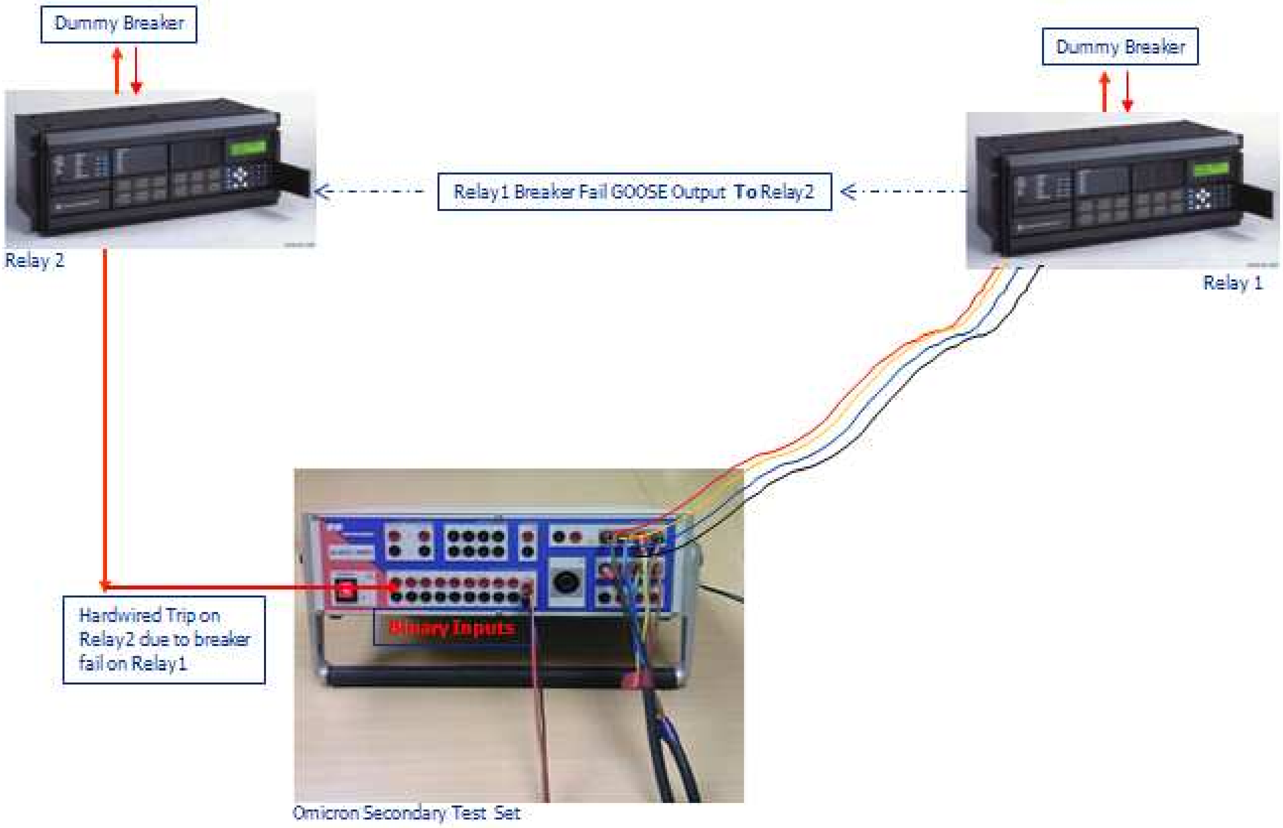


Figure 5.19 Setup for the circuit breaker fail operation

### 5.5.1 Methodology for experiment 3

Analogue values are injected by the Omicron Secondary test set; the currents are injected into the current inputs of relay-1 to simulate a fault condition on Feeder 1. Feeder 1 dummy breaker is connected to relay-1, therefore the dummy breaker is controlled by the relay to open and close.

To ensure that the circuit breaker fail functionality does operate for this experiment, the trip pulse on the dummy breaker is not wired to the Omicron injection test set to stop the injection of currents; relay-1 will therefore see it as a breaker fail. This will be seen as a breaker fail if the relay would have counted 140 milliseconds after it has sent a trip pulse to the breaker and currents are still flowing in the relay after this time. If the dummy breaker is wired to the Omicron test set, then the trip pulse from dummy breaker will stop the current injection from Omicron test set. This will result in breaker fail condition.

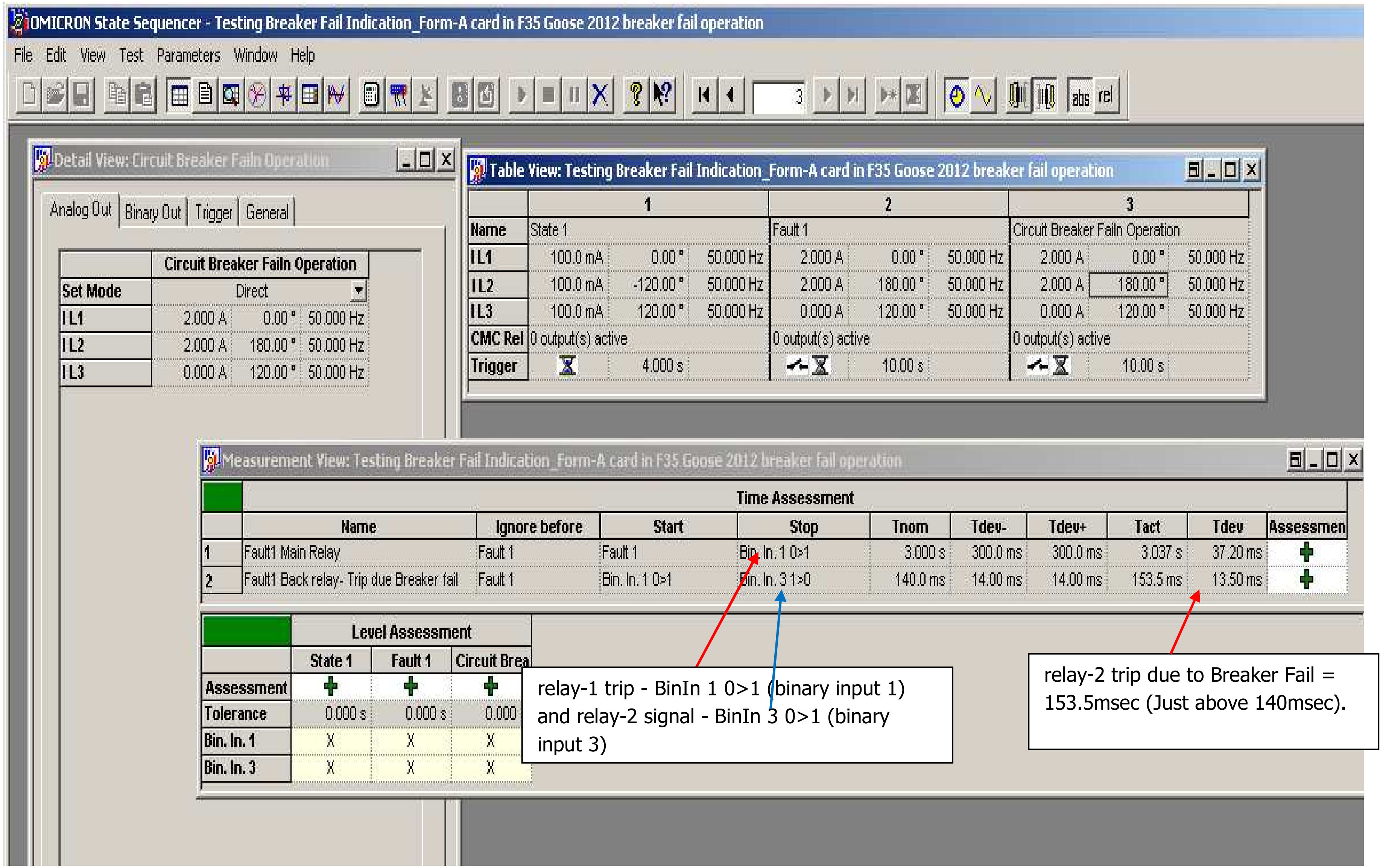
If the dummy breaker is not wired to the Omicron, the breaker fail will then be detected by relay-1 after 140 milliseconds. The breaker fail signal will then be sent to relay-2 via a GOOSE message. The GOOSE message will then be used in the logics of relay-2 to trigger the hardwired trip output to the Omicron test set and to the dummy breaker. This stops the currents input injection, indicating that the fault is being cleared. For GOOSE messages to be transmitted from one device to another, the message is either transmitted via an Ethernet fibre or copper cable and Ethernet switch. Figure 5.19 illustrates how the devices are connected for communication. The results for experiment 3 are shown in Figure 5.20.

According to Figure 5.20, there is a trip signal for Fault 1 (row-1) from relay-1; the trip signal is seen after 3.037 seconds after the fault inception. In row-1 the trip signal from relay-1 is indicated by BinIn 1 0>1 (binary input 1), and in row-2 the signal from relay-2 is indicated by BinIn 3 0>1 (binary input 3). In row-2, the signal that is observed to have been triggered is called the breaker fail signal and it is triggered by relay-2 in 153.5 seconds after the trip signal from relay-1. The 153.5 seconds is the time difference between (binary input 1) and (binary input 3), which is the time difference between trip signal from relay-1 after the breaker fail signal from relay-2.

From Figure 5.20, it can be observed that relay-1 tried to clear the fault at 3.037 seconds after inception of the fault, but it failed to do so. Thereafter there was a breaker fail condition, which resulted in relay-2 tripping 153.5 seconds after relay-1 tripped. Tnom (Nominal Time) in Figure 5.20 indicated the pre-defined expected trip time and Tact is the actual trip time. Relay-1 was expected to trip after 3 seconds but it tripped after 3.037, which is still acceptable since it is within the tolerance time (the allowed time for the results to be different from the expected time), which is 300 milliseconds. Relay-2 was expected to be triggered 140 milliseconds after relay-1 tripped, but it was triggered after 153.5 milliseconds, which is still acceptable as it too was within the tolerance time.

Experiment-3 is concluded to be successful since the results are as expected.

University of Cape Town



OMICRON State Sequencer - Testing Breaker Fail Indication_Form-A card in F35 Goose 2012 breaker fail operation

Detail View: Circuit Breaker Failn Operation

| | Circuit Breaker Failn Operation | | |
|---|---|---|---|
| Set Mode | Direct | | |
| I L1 | 2.000 A | 0.00 ° | 50.000 Hz |
| I L2 | 2.000 A | 180.00 ° | 50.000 Hz |
| I L3 | 0.000 A | 120.00 ° | 50.000 Hz |

Table View: Testing Breaker Fail Indication_Form-A card in F35 Goose 2012 breaker fail operation

| | 1 | | | 2 | | | 3 | | |
|---|---|---|---|---|---|---|---|---|---|
| Name | State 1 | | | Fault 1 | | | Circuit Breaker Failn Operation | | |
| I L1 | 100.0 mA | 0.00 ° | 50.000 Hz | 2.000 A | 0.00 ° | 50.000 Hz | 2.000 A | 0.00 ° | 50.000 Hz |
| I L2 | 100.0 mA | -120.00 ° | 50.000 Hz | 2.000 A | 180.00 ° | 50.000 Hz | 2.000 A | 180.00 ° | 50.000 Hz |
| I L3 | 100.0 mA | 120.00 ° | 50.000 Hz | 0.000 A | 120.00 ° | 50.000 Hz | 0.000 A | 120.00 ° | 50.000 Hz |
| CMC Rel | 0 output(s) active | | | 0 output(s) active | | | 0 output(s) active | | |
| Trigger | ⧗ | 4.000 s | | ⧗ | 10.00 s | | ⧗ | 10.00 s | |

Measurement View: Testing Breaker Fail Indication_Form-A card in F35 Goose 2012 breaker fail operation

| | Time Assessment | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Name | Ignore before | Start | Stop | Tnom | Tdev- | Tdev+ | Tact | Tdev | Assessmen |
| 1 | Fault1 Main Relay | Fault 1 | Fault 1 | Bin. In. 1 0>1 | 3.000 s | 300.0 ms | 300.0 ms | 3.037 s | 37.20 ms | + |
| 2 | Fault1 Back relay- Trip due Breaker fail | Fault 1 | Bin. In. 1 0>1 | Bin. In. 3 1>0 | 140.0 ms | 14.00 ms | 14.00 ms | 153.5 ms | 13.50 ms | + |

| | Level Assessment | | |
|---|---|---|---|
| | State 1 | Fault 1 | Circuit Brea |
| Assessment | + | + | + |
| Tolerance | 0.000 s | 0.000 s | 0.000 |
| Bin. In. 1 | X | X | X |
| Bin. In. 3 | X | X | X |

relay-1 trip - BinIn 1 0>1 (binary input 1) and relay-2 signal - BinIn 3 0>1 (binary input 3)

relay-2 trip due to Breaker Fail = 153.5msec (Just above 140msec).

**Figure 5.20 Breaker fail operation results from Omicron test set recordings**

# 6. Conclusions and Recommendations

## 6.1 Conclusions from the results and analysis:

1) From experiments 1.1a to 1.1c, where both relay-1 and relay-2 have used Form-C contact input and output cards with each relay using a debounce time of 16 milliseconds, the time difference between relay-2 GOOSE and the relay-2 physical contact output is 6.7 milliseconds. When the debounce of the relay-2 is reduced to 4 milliseconds, the relay-2 GOOSE output is found to be slower than the physical output by (-)3.1 milliseconds. Furthermore, when the debounce time of relay-2 is increased from 4 to 4.5 milliseconds, the GOOSE output becomes faster than the physical contact output by 800 microseconds.

   From the above results it can be concluded that:

   - The transmission of GOOSE messages is faster than the signal sent via the physical contact output but only under certain conditions. The results differ due to the operating and debounce time values. When the combination of debounce time of contact input and operating time of the contact output for relay-2 is 12 milliseconds (or less) in total, the GOOSE output becomes slower than physical contact output. Therefore, both the debounce time of the contact input and the operating time of the contact output of the relays have an impact on the performance of GOOSE messages.

2) For experiments 1.2a to 1.2c, where relay-1 is using Form-C contact input and output card and relay-2 is using Form-A contact input and output card, with each relay set to a debounce time of 16 milliseconds; the time difference between relay-2 GOOSE and the relay-2 physical contact output is 8.5 milliseconds. When the debounce of the relay-2 is reduced to 8 milliseconds, the relay-2 GOOSE output is found to be slower than physical contact output by 1.4 microseconds. Furthermore, when the debounce time of relay-2 is increased to 8.5 milliseconds, the GOOSE output becomes faster than physical contact output by 500 microseconds.

   From (2) above it can be concluded that:

   - Similar to experiment 1.1a to 1.1c, the transmission of GOOSE messages are faster than the signal sent via the physical contact output, but not always.

   - When the combination of debounce time of contact input and an operating time of the contact output for relay-2 is 12 milliseconds (or less) in total, the GOOSE output is slower than physical contact output. Therefore in this case again, the debounce time and the operating time have an impact in the performance of GOOSE messages.

   - This shows that the performance of GOOSE messages is dependent on factors such as the debounce time of contact input and the operating time of the output

University of Cape Town

contact of the relays, meaning one cannot assume that the transmission of GOOSE is always faster than that of the physical contact signals of relays/IEDs.

3) When the sampled values traffic is introduced there is difference in the results. This is according to the three tests that were performed in experiments 2.1 to 2.3. The impact is not very significant though because in each test, if the GOOSE output was faster than the physical contact output in the experiment where there was no sample values traffic, the GOOSE is still faster when the sampled values traffic are introduced. If GOOSE output was slower before sampled values traffic was introduced, it is still slower when sampled values traffic is introduced.

4) The non-conventional breaker fail bus was achieved without using a bus zone panel as an interface. The advantage of not using a bus zone interface panel is that less equipment and wiring are used by the substation.

   For this non-conventional breaker fail method, the GOOSE message that is carrying the breaker fail signal was sent from the relay-1 to relay-2. In this experiment, relay-1 is the relay that is experiencing a breaker fail condition and relay-2 is subscribed to it (relay-1) so that when the breaker fail occurs the GOOSE messages can be sent to it (relay-2).

This is an indication that GOOSE messages can be used to perform important operations such as breaker fail successfully, resulting in less equipment and wiring used in the substation, thus saving time and money for the users.

5) Interoperability (which is the ability of devices from different suppliers to communicate or exchange data) was implemented. F35 relays (products from General Electric) were able to communicate with the Omicron secondary inject test set (product from Omicron). The GOOSE messages were sent from the F35 relays to the Omicron secondary injection test set and the Omicron test set responded successfully.

University of Cape Town

## 6.2 Recommendations for Further Studies

- More tests should be performed with the relays of other vendors or suppliers to prove the performance of GOOSE messages. This is necessary since the GOOSE performance was shown to depend on the physical contact input/output that is installed in the relays; this includes the debounce time of contact inputs and the operating time of contact outputs.

- More tests should be performed with various levels of sampled values traffic to see the impact of sampled values traffic on the performance of GOOSE messages.

- Interoperability was implemented. More tests should be performed to demonstrate interoperability between IEDs/relays from different vendors or suppliers.

# Appendix A:

## Experiment 1.2a:

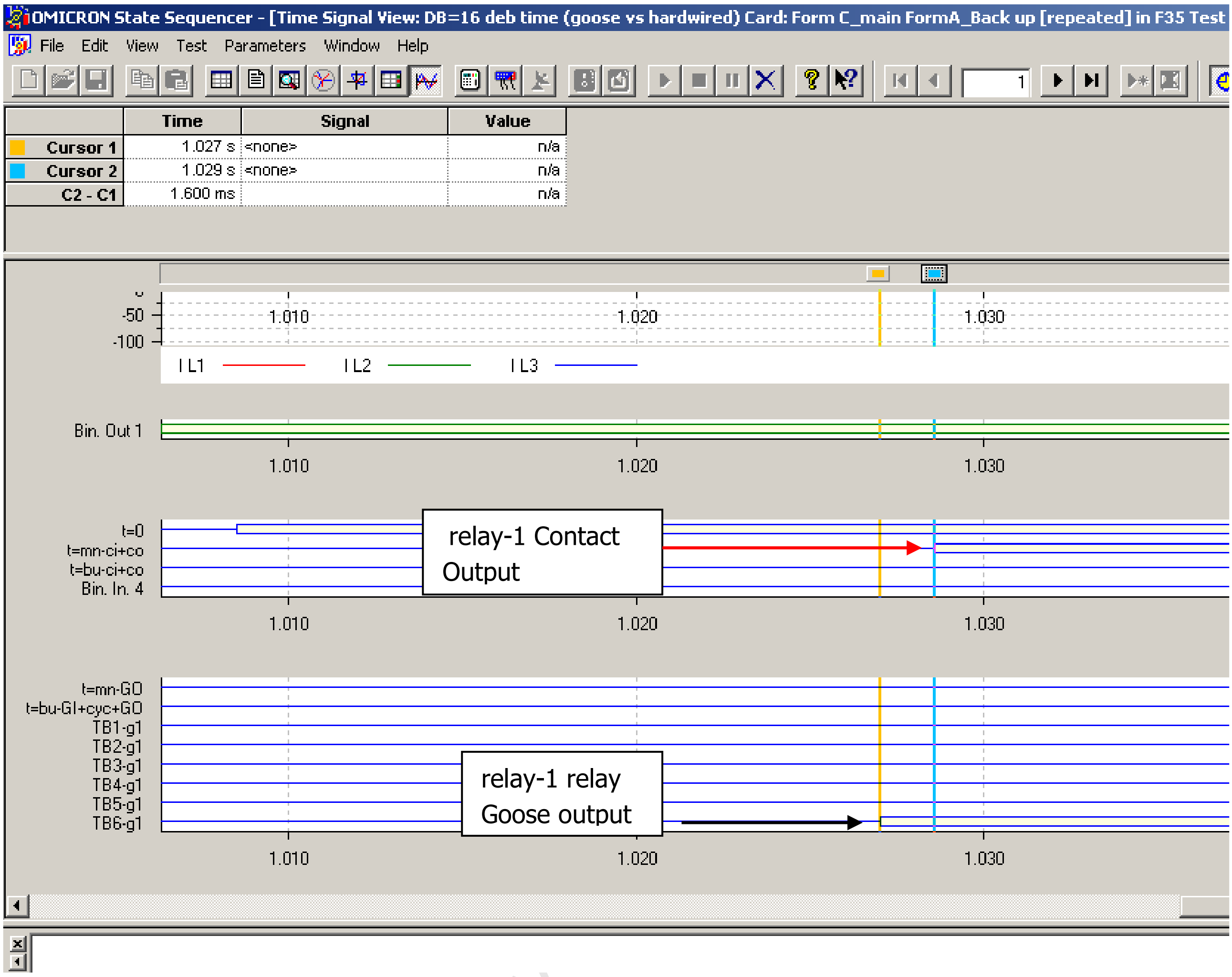
For experiment 1.2a contact inputs and outputs Form-C card is used on relay-1 while contact inputs and outputs Form-A card is used for relay-2. The Form-C cards contact output operation time is within 8 milliseconds, while the operating time of Form-A cards is within 4 milliseconds. The debounce time of contact inputs of both the relay’s cards are set to 16 milliseconds.

The experiment is as follows:

The experiment is started by closing the contact output of omicron test that is wired to the relay-1 physical contact input. The output of relay-1 goes out as both physical contact output and the GOOSE output simultaneously and independently.

Both F35 relay-1 physical contact output and GOOSE outputs are configured to the inputs of relay-2, which are physical contact input and GOOSE input of relay-2 respectively. The inputs of relay-2 are configured to its contacts outputs; these outputs of relay-2 are therefore asserted by relay-2 inputs. The results for experiment 1.2a are indicated from Figures A.1 to Figure A.2 and are as follows:

University of Cape Town



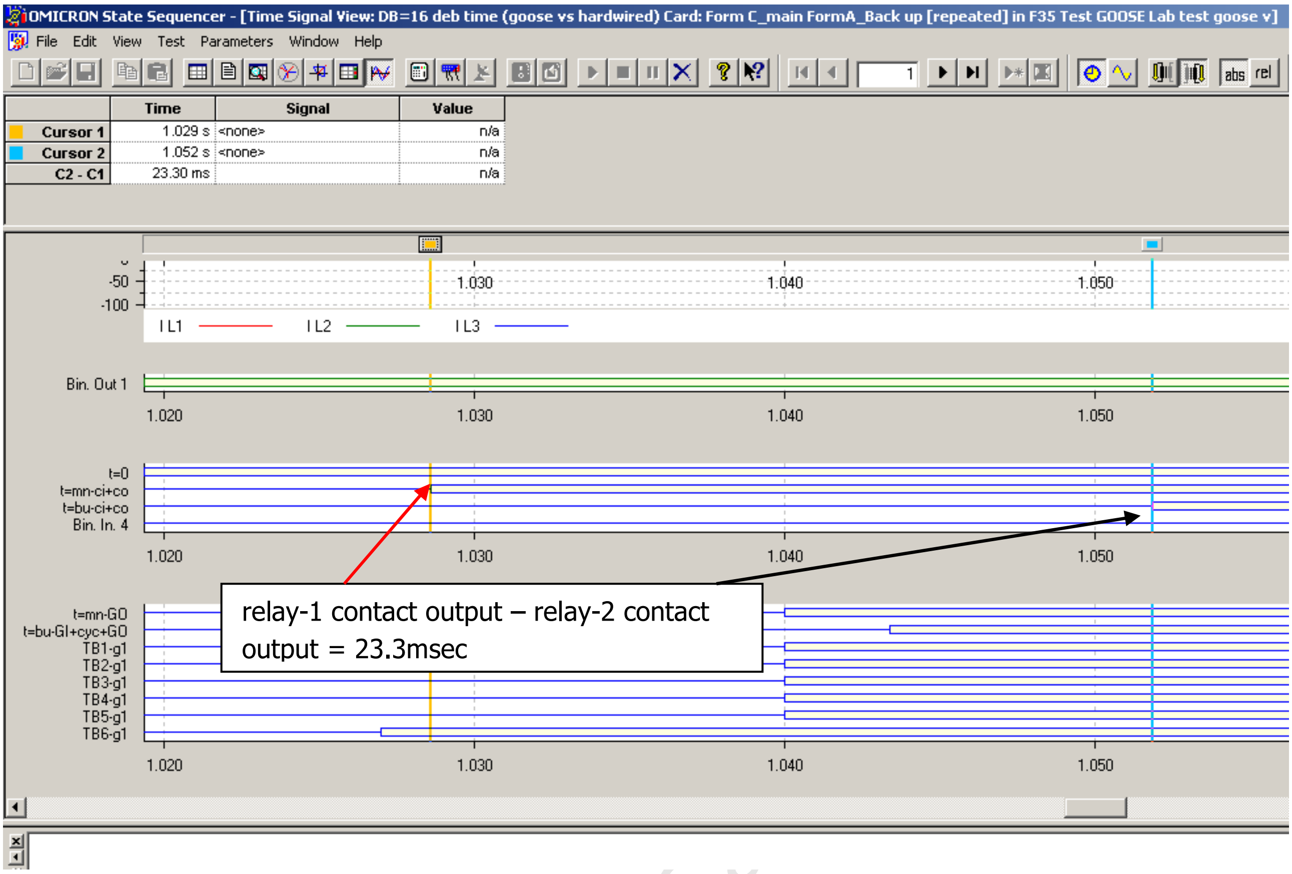
| | Time | Signal | Value |
|---|---|---|---|
| 🟨 **Cursor 1** | 1.027 s | <none> | n/a |
| 🟦 **Cursor 2** | 1.029 s | <none> | n/a |
| **C2 - C1** | 1.600 ms | | n/a |

**Figure A.1 Time difference versus the relay-1 physical output and GOOSE output**

Figure A.1 illustrates that relay-1 GOOSE output was triggered faster than the physical output contact. GOOSE labelled “TB6- g1” which is a Trip GOOSE (yellow line) is faster than the relay-1 physical contact output (blue line) labelled “t = mn-ci+co” by 1.6 milliseconds.

Figure A.2 illustrates the time difference between relay-1 physical contact output and relay-2 physical contact output.

University



|  | Time | Signal | Value |
|---|---|---|---|
| 🟧 Cursor 1 | 1.029 s | <none> | n/a |
| 🟦 Cursor 2 | 1.052 s | <none> | n/a |
| C2 - C1 | 23.30 ms |  | n/a |

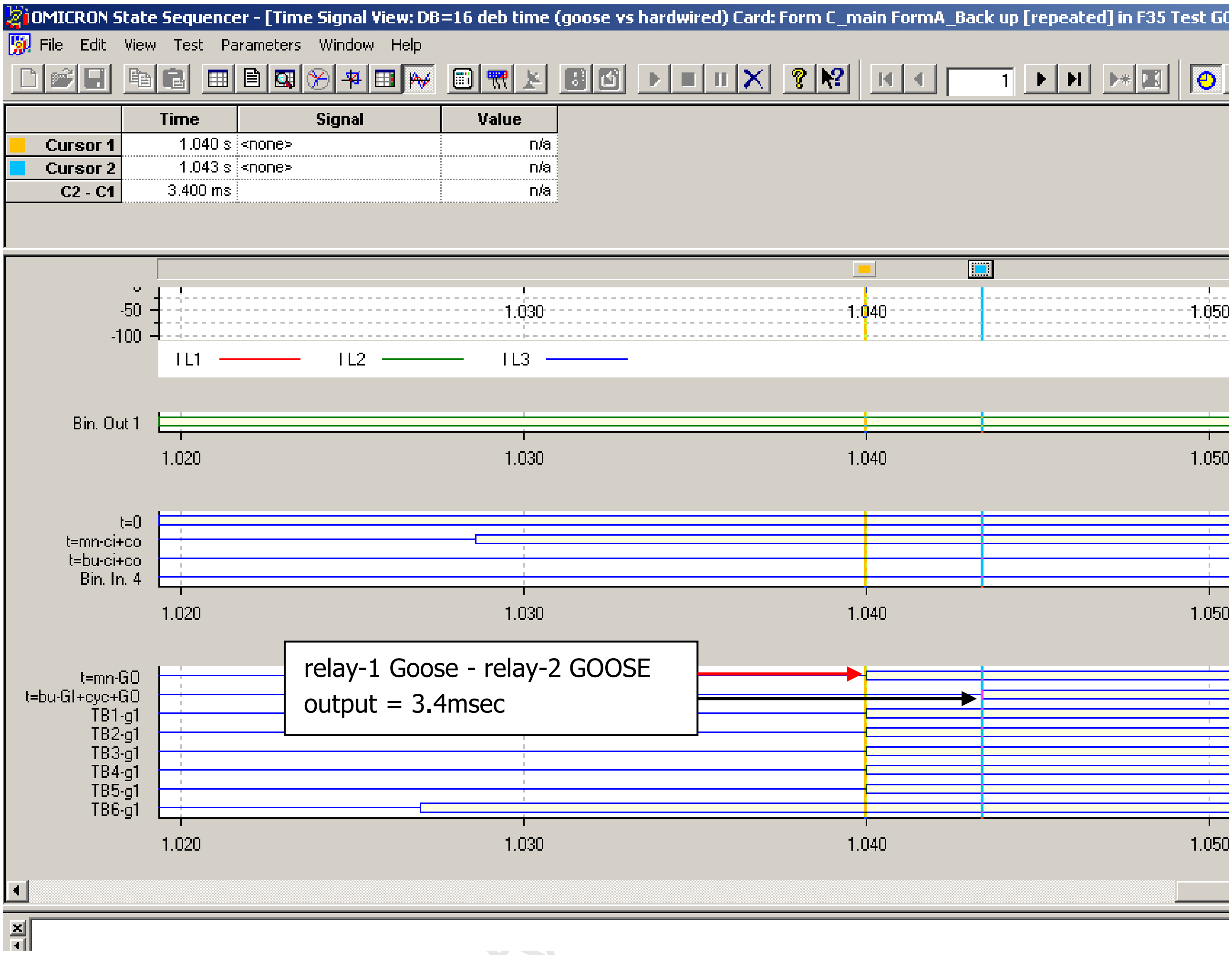
relay-1 contact output – relay-2 contact output = 23.3msec

**Figure A.2 Time difference between the relay-1 physical contact output and relay-2 contact outputs**

In Figure A.2, it can be observed that the time difference between relay-1 physical contact output and relay-2 physical contact output is 23.30 milliseconds. This time difference is about the expected time of 24 milliseconds, since the relay-2 contact input debounce time is set to 16 milliseconds and the physical contact output operates within 4 milliseconds.

Figure A.3 illustrates the time difference between relay-1 and relay-2 GOOSE outputs.

University of C



| | Time | Signal | Value |
|---|---|---|---|
| 🟧 Cursor 1 | 1.040 s | <none> | n/a |
| 🟦 Cursor 2 | 1.043 s | <none> | n/a |
| C2 - C1 | 3.400 ms | | n/a |

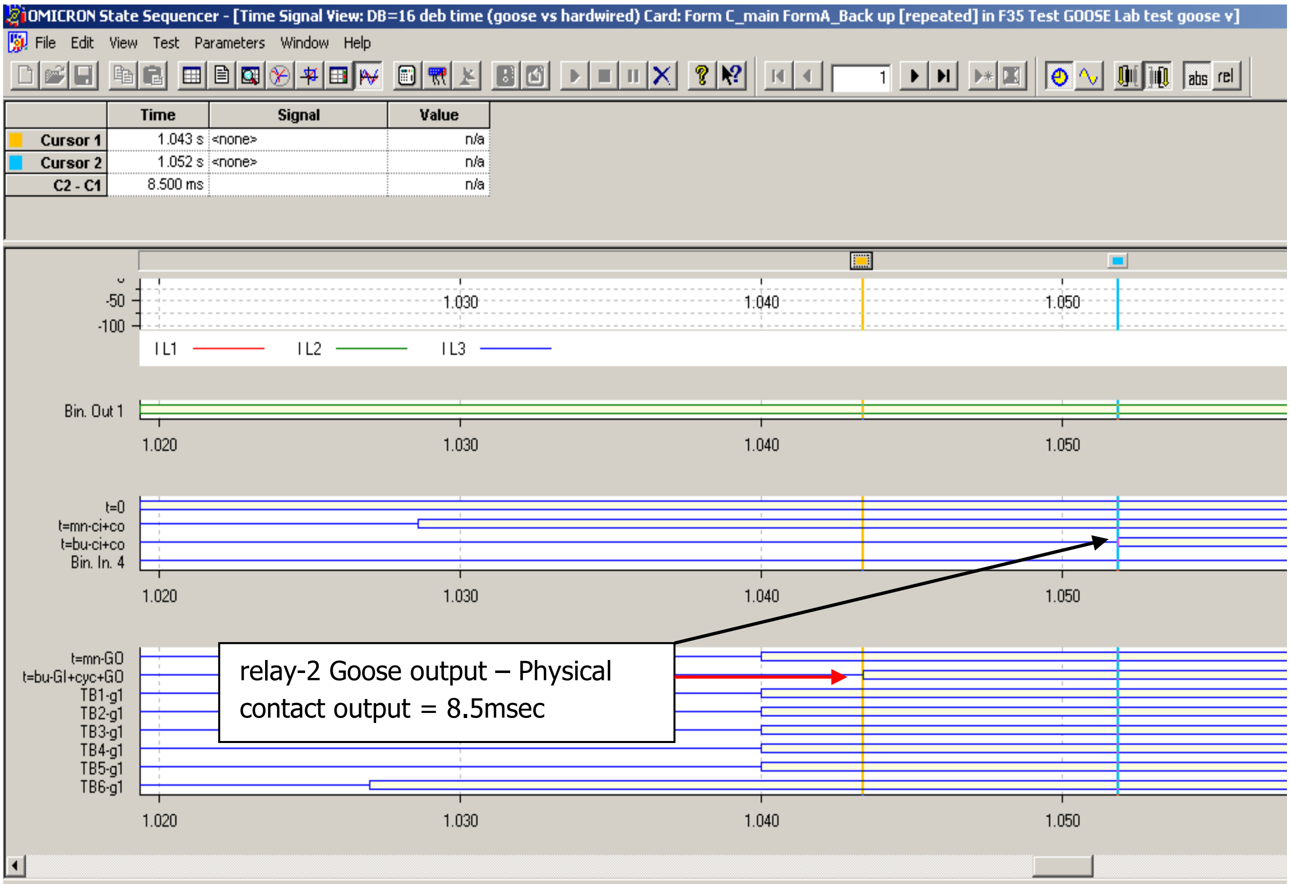
relay-1 Goose - relay-2 GOOSE output = 3.4msec

Figure A.3 Time difference between the relay-1 and relay-2 GOOSE outputs

In Figure A.3, it can be observed that the time difference between the relay-1 and relay-2 GOOSE outputs is 3.4 milliseconds.

Figure A.4 illustrates the time difference between relay-2 physical contact and GOOSE outputs.

University



| | Time | Signal | Value |
|---|---|---|---|
| Cursor 1 | 1.043 s | <none> | n/a |
| Cursor 2 | 1.052 s | <none> | n/a |
| C2 - C1 | 8.500 ms | | n/a |

relay-2 Goose output – Physical contact output = 8.5msec

**Figure A.4 Time difference between the relay-2 contact output and GOOSE output**

According to Figure A.4, it can be observed that the relay-2 GOOSE output (yellow line) is faster than its physical contact output (blue line) by 8.7 milliseconds.

University of

# References:

[1] D.J. Dolezilek, S.Schweitzer and Schweitzer Engineering Laboratories, “Inc.Practical Applications of Smart Grid Technologies”, *Reliable Power,J.* Vol.2, Feb. 2011, pp 2-13

[2] M. Desjardine, P. Forsyth and R. Mackiewicz, “Real Time Simulation Testing Using IEC 61850”, presentated at the Power Systems Transients conference (IPST’07), 2007, pp. 1-5.

[3] MOXA Pty Ltd. IEC61850 Edition 2 and its impact Seminar. “Practical Session -Power System Flow & Case Study.” Canada Cape Peninsular University of Technology, Bellville, South Africa, Sep. 27 -29, 2011 (please do as above letters should not go below the numbers!-seee IEEE template.

[4] IEC Communication Network and Systems in Substations, Com*munication requirements for functions and device models*. IEC61850-1 , 2003.

[5] T.S. Sidhu and Y.Yin. “Modelling and Simulation for Performance Evaluation of IEC61850-Based Substation Communication Systems”, *IEEE Trans. Power Delivery*, vol.22, pp.1482-1488, Jul. 2009.

[6] *Comptia Network+, Exam guide*, Mc Graw-Hill , M.Meyers, N Y, USA, 2009, pp 11-209.

[7] Gtcc-it, (2012, October 14) “OSI model communication” [Online]. Available at: www.gtcc-it.net/billings/anotherosi.htm

[8] Computer Hope (2013, January 22), “encapsulation.” [Online]. Available at: www.computerhope.com/jargon/e/encapsul.htm

[9] Wikipedia (2012, September 25), “Network Topology.”[Online] .Available at: http://en.wikipedia.org/wiki/Network_topology

[10] Wiki and Eghs. (2012, October 1) “Wide Area Network.” [Online]. Available at: http://itgsonline.wetpaint.com/page/NQ+-+Lan%2FWan

[11] Edraw Professional Diagram solution (2012, November 25) “Network Topology.” [Online]. Available at : www.edrawsoft.com/Network-Topologies.php

[12] Escotal (2012, October 16). “Star topology.” [Online]. Available at: www.escotal.com/Images/Network%20parts/topology_star.gif

[13] Escotal (2012, October 16). “Bus topology.” [Online]. Available at: www.escotal.com/Images/Network%20parts/topology_bus.gif

[14] Wordpress. (2012, October 16) “Mesh topology.” [Online]. Available at: http://eta305.files.wordpress.com/2007/08/topology_mesh.gif,

University of Cape Town

[15] Compnetworking. (2012, October 16) “Tree Topology.” [Online]. Available at: http://0.tqn.com/d/compnetworking/1/0/0/a/topology_tree.gif

[16] Exforsys Inc. (2013, January 27) “tree topology”, [Online]. Available at: www.exforsys.com/tutorials/networking/tree-topology.html

[17] Compnetworking (2012, October 16) “Ring topology.” [Online]. Available at: http://0.tqn.com/d/compnetworking/1/0/p/1/topology_ring.gif

[18] D. Davis. (2009,January 7) “Spanning Tree Protocol.” [Online]. Available at: www.petri.co.il/csc_preventing_network_loops_with_stp_8021d.htm

[19] A.Apostolov, “IEC 61850 based substation-to-substation communications.” presented at the 2011 Int. Conf. Cape Peninsular University of Technology IEC 61850 edition-2 and its impact Seminar, South Africa, Bellville.

[20] Wikimedia.( 2012, October 17) “Server based network.” [Online]. Available at: http://www.wikimedia.org

[21] Mscd.edu.(2012, October 17) “Switched peer to peer communication.”[ Online ]. Available at: http://clem.mscd.edu/~morrisgj/cms323/Images/Peer-to-Peer-new.Switch.gif

[22] IEC *Communication Network and Systems in Substations, Part 1 Introduction and Overview.* Switzerland. IEC61850 Standard, 2003.

[23] Wikipedia (2013, March 28) “GSSE” [Online]. Available at: http://en.wikipedia.org/wiki/Generic_Substation_Events

[24] IEC *IEC Communication Network and Systems in Substations, Communication requirements for functions and device models”.* Switzerland IEC61850 Standard ,2003.

[25] M. Rouse. (2005, September 25) “Debouncing.” [Online]. Available at: http://whatis.techtarget.com/definition/debouncing

[26] Wikipedia (2013, March 11). “Inter-locking.” [Online]. Available at: http://en.wikipedia.org/wiki/Interlock

[27] M. Everton. “Line Distance Protection up to including 132 kV.” S.A. Patent SCSAGAAI0, October 16, 2003.

[28] H. Kirrmann. “HSR – High Availability Seamless Redundancy Fault-tolerance in Ethernet networks IEC 62439-3”. Switzerland, Patent IEC SC65C WG15 (HA), August 17, 2010.

[29] V. Beal ( 2010, August 31 )“Router” [Online]. Available at: http://www.webopedia.com/DidYouKnow/Hardware_Software/2005/router.asp

University of Cape Town

[30] Dan (2007, November 11) “Switch” [Online]. Available at: http://answers.yahoo.com/question/index?qid=20071212192700AA9V7O4

[31] R.Hariram. “Substation Automation – Network Application Design Guide”. S.A. Patent TGL_41 684, Aug, 2012.

[32] J. Rinaldis (2013, May 6). “IEEE1588 message” [Online]. Available at: http://www.rtaautomation.com/ieee1588/

University of Cape Town