UNDERSTANDING MICROPROCESSOR-BASED TECHNOLOGY APPLIED TO RELAYING

Power System Relaying Committee

Report of Working Group I-01
of the
Relaying Practices Subcommittee

Members of the Working Group

Mohindar S. Sachdev, Chair
University of Saskatchewan
Saskatoon, SK, Canada
sachdev@ieee.org

Ratan Das, Vice Chair
ABB Inc.
Allantown, PA, USA
ratan.das@ieee.org

Members:

A. Apostolov
S. M. Brahma
A. Darlington
F. Lopez
M. Meisinger
D.S. Ouellette
T.S. Sidhu
J. Soehren
D.A. Tziouvaras
J. Uchiyama
J. Wang

Members of the Working Group I-16 which prepared the original report in 2004:

Mohindar S. Sachdev, Chair **Ratan Das, Vice-Chair**

A. Apostolov	J. Holbach	T.S. Sidhu
J.C. Appleyard	B. Kasztenny	J. Soehren
K. Behrendt	T.C. Lanigan	G. Swift
O. Bolado	P.G. McLaren	M. Thompson
G.A. Brunello	R.W. Patterson	D.A. Tziouvaras
J.R. Cornelison	M.M. Saha	

The following were the members of the Relaying Practices Subcommittee of the Power System Relaying Committee when this report was prepared.

Sidhu, T.S., Chair **Beresh, R., Vice-Chair**

Anderson, B.B.	Huddleston III, J.D.	Ranieri, M.M.
Apostolov, A.P.	Ingleson, J.W.	Sachdev, M.S.
Bajpai, M.	Jackson, B.	Sevcik, D.
Bolado, O.	Jodice, J.	Simon, M.S.
Boyle, J.R.	Kezunovic, M.	Skendzic, V.
Burnworth, J.	Kojovic, L.	Smith, L.E.
Carpenter, W.M.	Kotheimer, W.C.	Swanson, M.J.
Chadwick Jr., J.W.	Kotos, P.A.	Tengdin, J.T.
Das, R.	Lowe, W.G.	Thorp, J.S.
Drum, P.R.	Makki, A.	Tziouvaras, D.A.
Friend, F.	McLaren, P.G.	Udren, E.A.
Gilbert, J.G.	Meisinger, M.	Usman, S.M.
Gilleland, J.G.	Moskos, G.P.	Weers, D.
Hackett, J.	Mugalian, B.	Whatley, J.
Hasenwinkle, I.O.	Nemier, M.	Whittaker, R.
Higinbotham, W.	Pettigrew, R.D.	Young, R.
Holbach, J.	Pickett, B.A	Zocholl, S.E.
Horowitz, S.H.		

Table of Contents

1 INTRODUCTION

1.1 Brief History of Microprocessor-based Relays

Different technologies have been used in the past to implement protection functions that properly detect disturbances in power systems and initiate the disconnection of the faulted components.

Originally, electromechanical relays were used to protect power systems. Most relays used either electromagnetic attraction or electromagnetic induction principle for their operation. Plunger type (electromagnetic) relays formed instantaneous units for detecting overcurrent or over-voltage conditions. Balanced-beam relays provided differential protection, distance protection as well as low burden (power requirement) overcurrent units. These relays operated when the magnitude of an operating signal was larger than the magnitude of the restraining signal. These relays were classified as amplitude comparators.

Single input induction type relays provided operations with time delays. Two-input induction type relays provided directional protection. Two- and three-input induction type relays also provided distance protection. The operation of these relays depended on the phase displacement between the applied electrical inputs. These relays were classified as phase comparators.

When solid-state technology was introduced, amplitude and phase comparison were implemented using discrete components including vacuum tubes. In early 1960's, advances in the integration of electronic circuits made this technology suitable for use in relays. The major advantage of these relays was that no moving parts were needed for performing their intended functions. The operating speeds of these relays were also more than the speed of their electromechanical counterparts and, their reset times were less than the reset times of their electromechanical counterparts. In addition to these benefits, the solid-state relays could be set more precisely and needed less maintenance.

Solid-state relays appeared to be the technology poised to replace the electromechanical counterparts in late 1960's when researchers ventured into the use of computers for power system protection. Their attempts and the advances in the Very Large Scale Integrated (VLSI) technology and software techniques in the 1970's led to the development of microprocessor-based relays that were first offered as commercial devices in 1979. Early designs used the fundamental approaches that were previously used in the electromechanical and solid-state relays.

In spite of the developments of complex algorithms for implementing protection functions, the microprocessor-based relays marketed in 1980s did not incorporate them. Those relays performed basic functions, took advantage of the hybrid analog and digital techniques, and offered good economical solution. The performance of these relays, however, was only adequate but their introduction did a lot of good to the highly conservative world of power system protection. Continuous advances in electronics, combined with extensive research conducted in microprocessor-based systems, led to a few applications in which multiple functions were performed by a microprocessor relay.

Electromechanical relays had no significant drawbacks in their protection functions, but the additional features offered by the microprocessor technologies encouraged the evolution of relays that introduced many changes to the industry. Despite their advantages, the new technology was competing with well-proven devices. Economics and additionally, functionality were probably the main factors that forced the industry to accept and cope with the changes.

Multifunction relays were introduced in the market in the late 1980s. These devices reduced the product and installation costs drastically. This trend has continued until now and has converted microprocessor relays to powerful tools in the modern substations.

At this time, several trends are emerging. These include common hardware platforms, configuring the software to perform different functions, integrating protection with substation control, and substituting cables carrying voltages and currents with fiber optic lines carrying signals in the form of polarized light.

On the software side, artificial intelligence techniques, such as neural networks, and adaptive protection are some of the fields that are being applied in protection practices. Recent work includes feedback systems in which relays monitor the operating state of the power system and automatically reconfigure themselves for providing optimal protection.

1.2 Benefits of Microprocessor-based Relays

While the basic protection principles have remained essentially unchanged throughout the evolution of the microprocessor-based relays, the adoption of this technology has provided many benefits, and a few shortcomings, compared to the previous technologies. The benefits are discussed in this section. The emphasis is on comparing the microprocessor technology with the electromechanical technology. Comparison with the solid-state technology is also included as required.

As is the case in most evolutionary changes, the boundaries between different technologies are fuzzy. There exist solid-state and microprocessor-based relays that utilize essentially analog measurement techniques. There are also digital relays that are hardware-based as opposed to software-based. For the purposes of the discussion in this section, microprocessor relays are defined as relays that utilize software based numerical measuring techniques.

1.2.1 Multiple functions

Microprocessor relays provide many functions that were not available in electromechanical or solid-state designs. These features include multiple setting groups, programmable logic, adaptive logic, self-monitoring, self-testing, sequence-of-events recording, oscillography, and ability to communicate with other relays and control computers. While these features make the relays very powerful, they also introduce factors, such as complexity, that were not associated with earlier technologies.

1.2.2 Cost

The cost per function of microprocessor-based relays is lower compared to the cost of their electromechanical and solid-state counterparts. The reduction in cost is due to the lower cost of components, production equipment and production techniques. Because microprocessor-based relay designs incorporate commercially available electronic components, their product life is limited by the product life of the components. While this factor is not within the relay manufacturer's control, it has proven to be a significant problem with the solid-state as well as the microprocessor-based relays.

1.2.3 Custom logic schemes

A major feature of microprocessor-based relays that was not available in previous technologies is the ability to allow users to develop their own logic schemes, including dynamic changes in that logic. This benefit, however, comes at a cost because this capability increases the complexity of the system. If the internal logic is not documented with the same care as other settings, it could easily lead to user errors, especially if different logic schemes are provided for use by a relay for different applications.

1.2.4 Panel space

Microprocessor- based protection systems require significantly less panel space than the space required by electromechanical and solid-state systems that provide similar functions. The reduction in size is a result of the high level of integration of the hardware and the ability of using one physical device for performing multiple protection functions, such as, overcurrent and multiple zone distance relaying for phase and ground fault protection. The shortcoming of this benefit is that it increases the susceptibility to common mode failure of the protection schemes.

1.2.5 Burden on instrument transformers

Microprocessor-based relays place significantly less burden on instrument transformers than the burden placed by the relays of the previous technologies. When relays of the previous technologies were used, the ability to provide protective functions was limited by the burden that could be placed on instrument transformers. This is not the case when microprocessor-based systems are used. In addition, microprocessor-based relays can be programmed to detect saturation of instrument transformers for minimizing incorrect operations. They also may require fewer CT and PT connections because some operating quantities, such as zero sequence currents and voltages, are derived by numerical techniques.

1.2.6 Sequence of events and oscillography

Reporting features, including sequence of events recording and oscillography are a natural by-product of microprocessor-based protection systems. These features make it possible to better analyze the performance of relays as well as system disturbances at minimal additional costs.

1.2.7 Self-monitoring and self-testing

Another advantage of microprocessor-based relays is their ability to perform self-monitoring and self-testing functions. These features reduce the need for routine maintenance because the relays automatically take themselves out of service and alert the operators of the problem when they detect functional abnormalities.

1.3 Shortcomings of Microprocessor-based Relays

While microprocessor-based relays have several advantages, they also have a few shortcomings that are not directly offset by specific benefits. Major shortcomings are discussed in this section.

1.3.1 Short life cycle

Microprocessor-based devices, including the protection systems, have short life cycles. While each generation of microprocessor-based systems increases the functionality compared with the previous generation, the pace of change makes the equipment obsolete in shorter times. This makes it difficult for the users to maintain expertise in using the latest designs of the equipment.

Another variation of this shortcoming is in the form of changes in the software used on the existing hardware platforms. Sometimes, these changes effectively generate newer relay designs. This requires that a software tracking system be used for each device owned by a utility.

1.3.2 Susceptibility to transients

Electromechanical devices are inherently immune to electrical transients such as EMI, RFI, etc. Early designs of solid-state relays were susceptible to incorrect operations due to transients but later designs included adequate countermeasures. Because of a better understanding of the problems, the microprocessor-based protection systems were designed in a manner that provided excellent reliability under those conditions as long as they conform to the IEEE Std. C37.90 or IEC 61000 series of standards. However, microprocessor-based protection systems will always remain more susceptible to such problems because of the nature of the technology compared to the systems built with the electromechanical technology.

1.3.3 Settings and testing complexity

Many microprocessor-based relays, which are designed to replace the functions of several solid-state or electromechanical relays, offer programmable functions that increase the application flexibility compared with the fixed function relays. The multi-function microprocessor-based relays, therefore, have a significant number of settings. The increased number of settings may pose problems in managing the settings and in conducting functional tests. Setting-management software is generally available to create, transfer, and track the relay settings. Special testing techniques, specifically the ability to enable and disable selected functions, are generally used when microprocessor-based relays are tested. This increases the possibility that the desired settings may not be invoked after testing is completed. Proper procedures must be followed to insure that correct settings and logic are activated after the tests are completed.

1.4 General Comments

The evolution of microprocessor-based protection systems has not been without its challenges. The industry has, however, come to the conclusion that the benefits far outweigh the shortcomings. This has gradually increased the acceptance of microprocessor-based protection systems during the previous twenty years.

One fundamental factor has, however not changed. Misapplication of microprocessor protective relays has the same impact on protection reliability as the misapplication of relays of the prior technologies had.

1.5 Major Functional Blocks of a Typical Microprocessor Relay

The block diagram of a typical microprocessor-based relay is shown in Figure 1.1. This relay samples voltages and currents, which, at the power system level, are in the range of hundreds of kilo volts and kilo amperes respectively. The levels of these signals are reduced by voltage and current transformers typically to 67 V and 5 A nominal values in North America.

The outputs of instrument transformers are applied to the analog input subsystem of the relay. This subsystem electrically isolates the relay from the power system, reduces the level of the input voltages, converts currents to equivalent voltages and removes high frequency components from the signals using analog filters. The outputs of the analog input subsystem are applied to the analog interface, which includes amplifiers, multiplexers and analog-to-digital (A/D) converters.

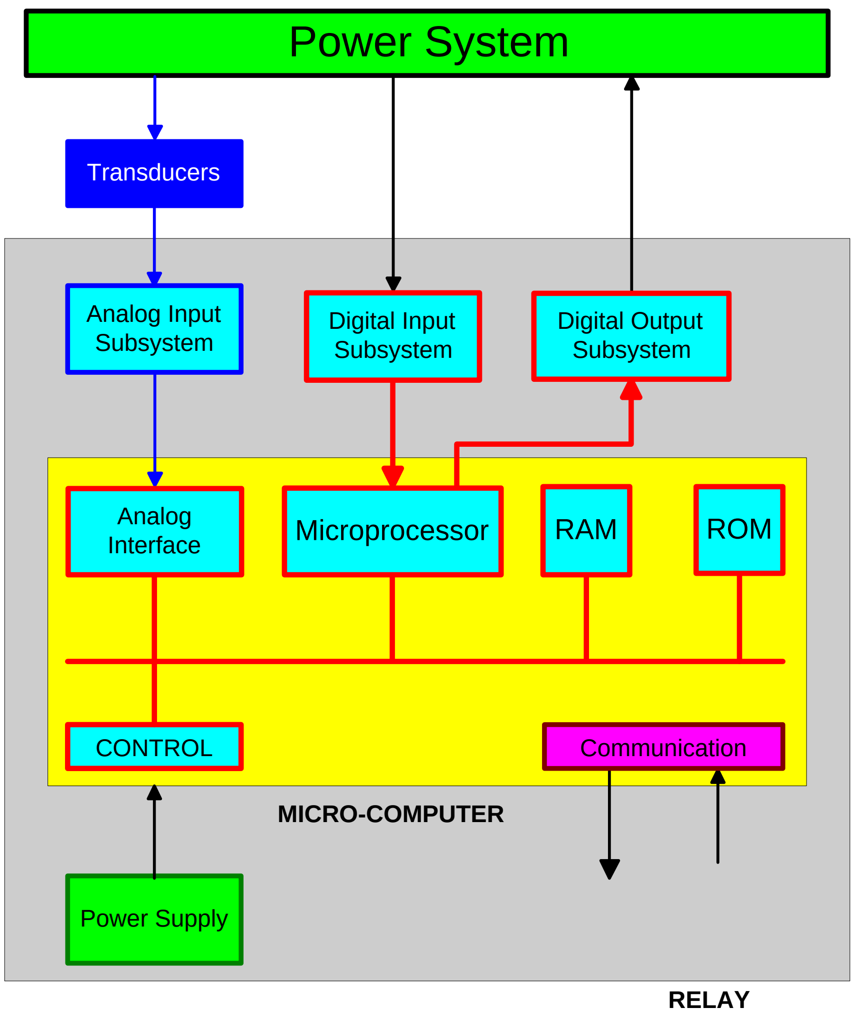


Figure 1.1: Block diagram of a typical microprocessor-based relay

These components sample the reduced level signals and convert their analog levels to equivalent numbers that are stored in memory. The status of isolators and circuit breakers in the power system is provided to the relay via the digital input subsystem and are read into the microcomputer memory.

A relaying algorithm, which is a part of the software, processes the acquired information. The algorithm uses signal-processing techniques to estimate the magnitudes and angles of voltage and current phasors. In some cases, the frequency of the system is also measured. These measurements are used to calculate other quantities, such as impedances. The computed quantities are compared with pre-specified thresholds (settings) to decide whether the power system is experiencing a fault/abnormal operating condition or not. If it is, the relay sends a command to open one or more circuit breakers for isolating the faulted zone of the power system. The trip output is transmitted to the power system through the digital output subsystem.

The relay settings and other vital information are stored in non-volatile memory of the relay. Random-access memory (RAM) is used for storing data temporarily. The power supply to a relaying microcomputer must be available even when the system supply is interrupted. Arrangements are, therefore, made to provide energy to the relay during normal and abnormal operating conditions of the power system.

Microprocessor-based relays are called numerical relays specifically if they calculate the algorithm numerically. The signal and data flows in these relays are shown in Figure 1.2. The relay is isolated from the power system by using auxiliary transformers which receive analog signals and reduce their levels to make them suitable for use in the relays. Since the A/D converters can handle voltages only, the currents are passed through precision resistors to convert them to voltages proportional to the currents.

During digital processing, high frequency components can appear to belong to the fundamental frequency class. This phenomenon is referred to as aliasing. To prevent aliasing from affecting the relaying functions, anti-aliasing filters (which are lowpass filters) are used along with the analog input isolation block.

After being quantized by the A/D converter, analog electrical signals are described by discrete values of the samples taken at specified instants of time. These discrete numbers are processed by using numerical methods. For example, quantized values of current and voltage samples are used to estimate the magnitudes and angles of their phasors. Voltage and current phasors are further used to calculate impedances as seen from a relay location.

The digital signals, also called binary or contact inputs, are applied to the relay via optic isolators that insure physical disconnection of the relay from the power system.

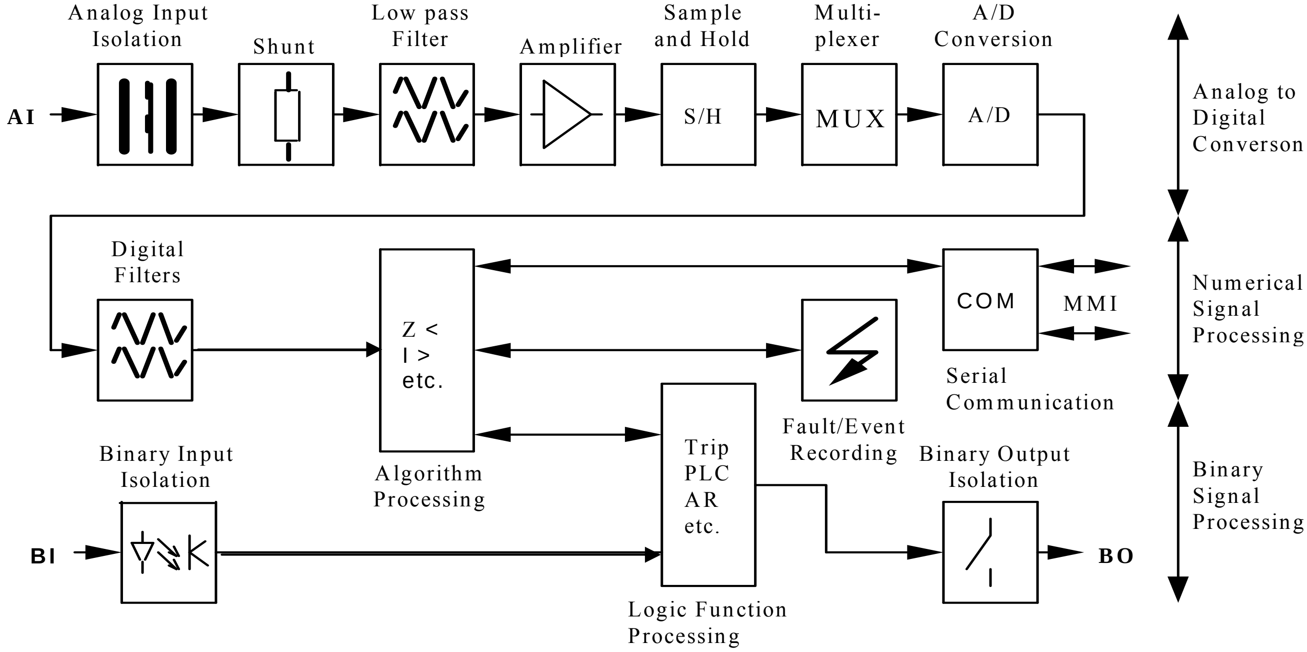


Figure 1.2: Signal flow diagram of a numerical relay

2 ANALOG INPUTS

This section describes the process of converting analog signals to sequences of numerical values. The need for pre-filtering is first discussed. The sampling and A/D conversion are then described.

2.1 Pre-filtering

The major component of a power system signal is its fundamental frequency component. However, harmonic components are also present to some levels due to non-linearities of the system and loads. Non-harmonic components are also present during faults; these are generated by traveling waves and their reflections from discontinuities.

2.1.1 Aliasing

It is necessary that an appropriate sampling rate should be used for converting analog signals to sequences of numbers because high-frequency components, which might be present in the signal, could be incorrectly interpreted as components of lower frequencies. For appreciating this phenomenon, consider a signal of 660 Hz shown in Figure 2.1. If this signal is sampled 600 times a second, the sampled values at different instants would be as shown in Figure 2.2. The reconstruction of the sampled sequence, and its interpretation by an algorithm, indicates that the signal is of the 60 Hz frequency. This misrepresentation of the high frequency component as a low frequency component is referred to as aliasing.

It can be concluded from this observation that, for obtaining a correct estimate of the component of a selected frequency, the sampling rate should be chosen in such a manner that components of higher frequencies do not appear to belong to the frequency of interest.

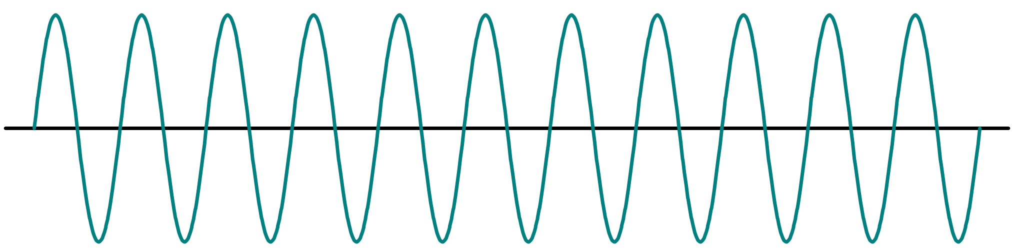


Figure 2.1: An ac voltage component of 660 Hz frequency

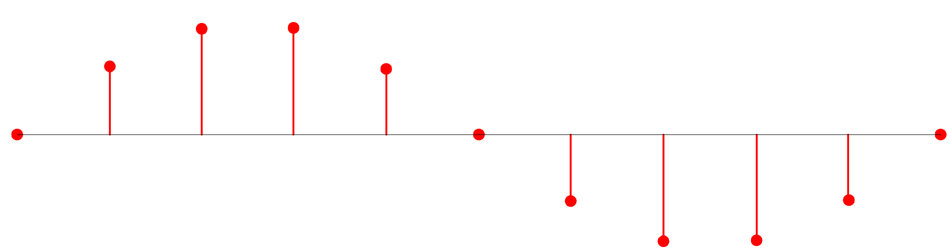


Figure 2.2: Samples of 660 Hz signal of Figure 2.1 taken at 600 samples per second

2.1.1 Minimizing aliasing

Because it is not always possible to choose a sampling rate that would prevent all high frequency components from appearing as components of the frequency of interest, the analog signals are applied to lowpass filters and their outputs are processed further. This process of band-limiting the inputs removes most of the high-frequency energy. If additional removal of high-frequency components is necessary, digital filtering within the relaying algorithm must provide it. The sampling rate and aliasing are intertwined by the Nyquist criterion.

2.2 Sampling

Sampling is the process of converting a continuous time signal, such as a current or voltage, to a discrete time signal. The sampling rate is usually set as high as is practical considering the capabilities of the microprocessor and A/D converter. The peak value of a sinusoidal waveform of a single frequency can be measured with a sampling rate as low as three samples in a period. The sampling rate of four samples per cycle was used in earlier relays when the capabilities of processors were not sufficient to handle data obtained by using higher sampling rates. More recent numerical relays use sampling rates that are as high as 96 samples per period.

A continuous signal is sampled properly if the samples represent the analog signal uniquely and contain enough information to recreate the original waveform. The recreation process is not as simple as drawing straight lines between the data points but, with the unique collection of numerical values of samples, it can be done.

A continuous waveform must be sampled at a minimum rate of twice the frequency of the component of highest frequency. This "rule" is usually referred to as the sampling theorem and is frequently called the Shannon sampling theorem, or the Nyquist sampling theorem. The following example tries to clarify the issue.

Five periods of a 60Hz analog signal are shown in Figure 2.3. A 300 Hz (5^{th} harmonic) signal whose amplitude is 20% of the amplitude of the 60 Hz signal is also shown in this figure. The composite waveform that contains these signals is shown in Figure 2.4. Sampling this waveform 960 times per second (16 samples per period of 60 Hz) and joining the sampled and quantized values by straight lines recreates the waveform shown in Figure 2.5. The Nyquist frequency for this sampling rate is 480 Hz (equal to the frequency of the 8^{th} harmonic). Because the highest frequency contained in the signal is less than the Nyquist frequency, this component is properly accounted for in subsequent processing.

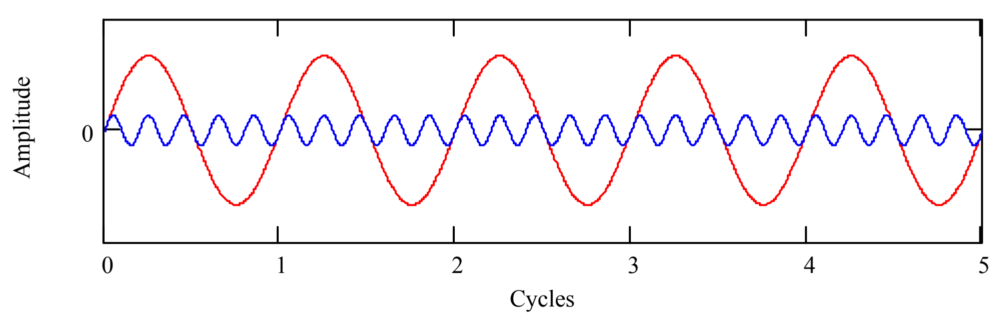


Figure 2.3: A 60Hz sinusoid along with a 20% 5^{th} harmonic sinusoid

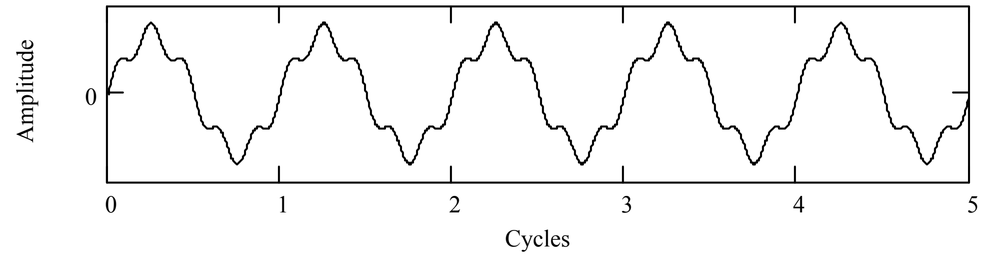


Figure 2.4: Waveform that contains the components of the sinusoids of Figure 2.3

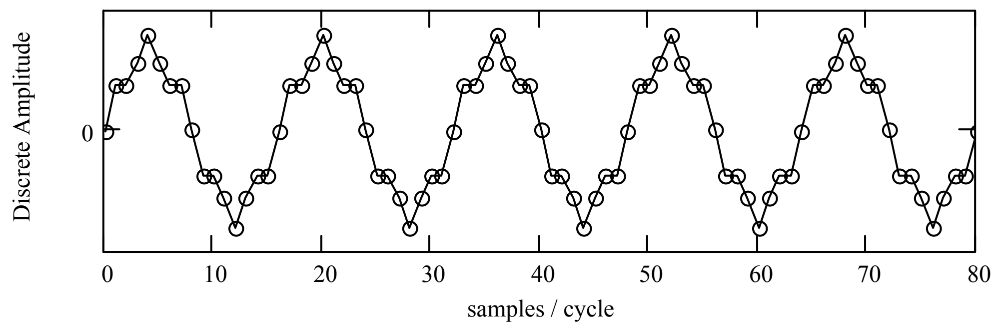


Figure 2.5: Waveform recreated by joining by straight lines samples taken at a rate of 960 samples per second

Sampling the signal 480 times per second and recreating the waveform provides the signal shown in Figure 2.6. The Nyquist frequency in this case is 240 Hz (equal to the frequency of the 4th harmonic). Because the Nyquist frequency is less than the frequency of the 5th harmonic that is present in the signal, this component is not present when the waveform is recreated and is not properly accounted for in subsequent processing. Distinctive "humps" seen in Figures 2.4 and 2.5 are missing from Figure 2.6. This is undesirable for two reasons; firstly, the information about the 5th harmonic has been lost, and secondly, the energy in the 5th harmonic component has corrupted the interpretation of the waveform. To avoid this problem, a pre-processing filter is used to remove the 5th harmonic before sampling.

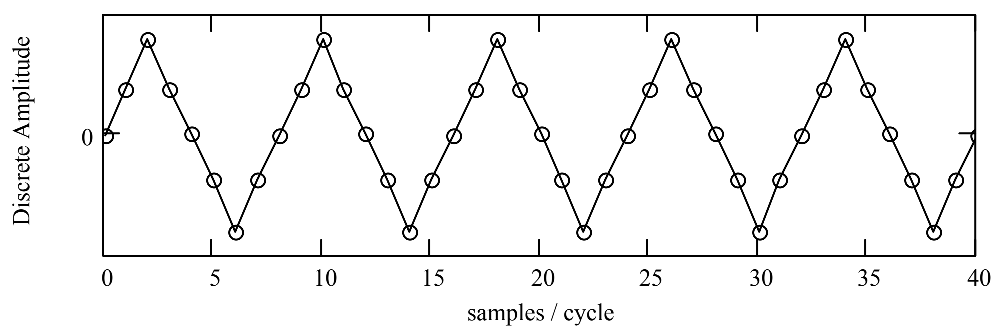


Figure 2.6: Samples taken at 480 samples per second and recreating the waveform by joining the samples by straight lines

2.3 Analog to Digital Conversion

Analog to digital (A/D) converters take the instantaneous values of the continuous time (analog) signal, convert them to equivalent numerical values and provide the numbers as binary outputs that represent the analog signal at the instants of sampling. Figure 2.7 shows a (n+1)-bit output of an A/D converter. As indicated in this word, one in the first bit is worth 2^0 (=1), one in the second bit is worth 2^1 (=2), one in the third bit is worth 2^2 (=4) and so on. The first bit in which one represent 2^0 is referred to as the least significant bit.

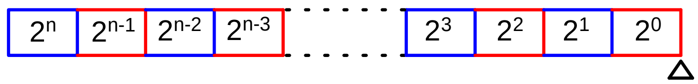


Figure 2.7: An n+1 bit output of an A/D converter

A properly converted 5V, 8-bit A/D converter would output a binary value 11111111 when the analog signal is 5 volts. This is the largest number that this converter can output that is equivalent to the decimal number 255 ($2^0+2^1+2^2+2^3+2^4+2^5+2^6+2^7 = 2^8-1$). . When the input is –5 volts, the output of the A/D converter would be a binary value 00000000 (0 decimal) and when sampled at zero volts, the output of the A/D converter would be 01111111 (127 decimal).

The numerical output has, therefore, 256 discrete levels. This means that the smallest level of the converter (binary 1) represents 39mV (10/256). This is 0.78 % of the peak value of the signal. A similar consideration shows that a 5 V, 16-bit A/D converter would have 56,536 discreet levels. This means that the Least Significant Bit (LSB) of the converter represents 0.15 mV ($10/2^{16}$). This is 0.0031 % of the peak value of the signal that converter can process without saturating. This is an increase in the resolution of the numerical output by a factor of 256.

Figure 2.8 shows two cycles of a 60 Hz analog waveform and the quantized values of the samples when the sampling rate is 960 samples per second. The “stair-case” trace is the sampled data that consists of discrete values whereas the analog waveform takes on a continuous range of values. The difference in magnitude between the consecutive samples taken near the zero crossings is much larger than the differences near the peaks of the waveform. This happens because the rate of change of the signal is greatest near the zero crossings.

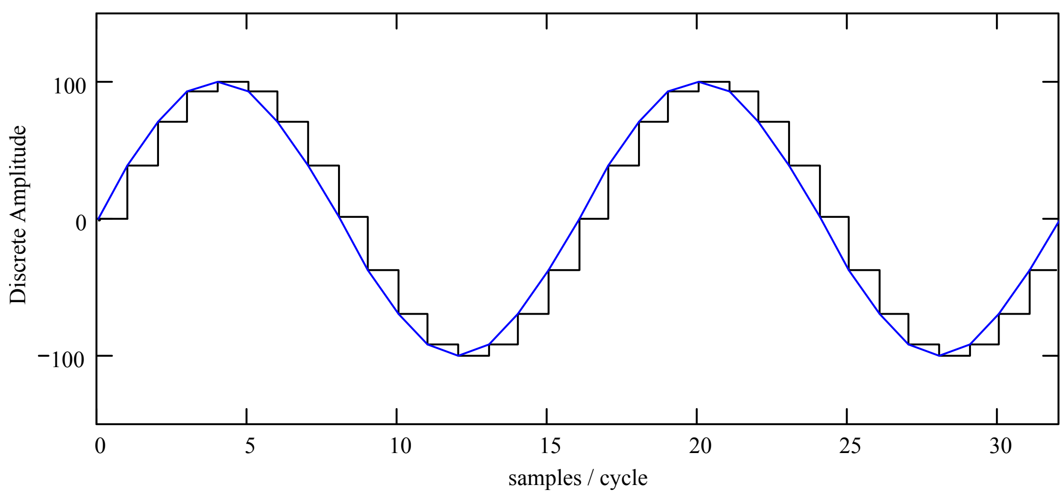


Figure 2.8: A 60 Hz waveform sampled at 960 samples per second

For each discrete step in Figure 2.7, the A/D conversion provides a binary value for use by the microprocessor, which should preferably complete the computation of the algorithm and logic before the next sample is taken. The job of the microprocessor expands considerably when many analog signals (ranging from three for overcurrent protection to 24 for a bus-differential application on a substation with six circuits connected to the bus) are used. The work of the microprocessor that has to be completed in one time step becomes significant in many applications. It, therefore, becomes necessary to use higher power microprocessors or develop "shortcuts" in processing the data.

The inputs are signals that have continuous levels but the outputs have limited levels. Most of the time, the output do not exactly match the inputs levels. Two options are available; the first option is that part of the signal that is less than the level of the least significant bit is discarded. This is called truncation. The second option is that a level of less that one-half of the least significant bit is ignored and a level of more than one-half and less than one significant bit is converted to one least significant bit. This is called rounding.

2.3.1 Analog to Digital converters

Since the introduction of the A/D converter technology, several types of converters have been developed. Some of the types are

- Counter Ramp Converter
- Single and Dual Ramp converters
- Tracking Converter
- Successive Approximation Converter
- Flash Converter

Successive approximation converters have been used in relays for a long time. Some relays now use flash converters. These two types of converters are described in this section.

A. General issues

An n-bit Analog to Digital converter (ADC) produces an n-bit binary number that can be manipulated by computing devices. The largest number of the n-bit word should represent the largest input voltage. The ratio of the binary number generated by the ADC to the largest number the binary work can have should be the same the ratio of the input voltage to the largest input voltage that the converter can handle. The conversion by an ADC may be considered as a two step process; first step is sampling and the second step is quantizing (generating the numerical value). Most converters include rounding so that the average of the numerical values generated by the converter is close to the average of the input signal and error due to quantizing is minimal. The numerical code generated by an ideal ADC with rounding as the input voltage increases and the error as a function of the input voltage are shown in Figure 2.8. As shown in this figure, the errors vary from -0.5 LSB to +0.5 LSB. In practice, a commercial ADC does not perform ideally. Consider a three-bit ADC whose conversions are as shown in Figure 2.9.

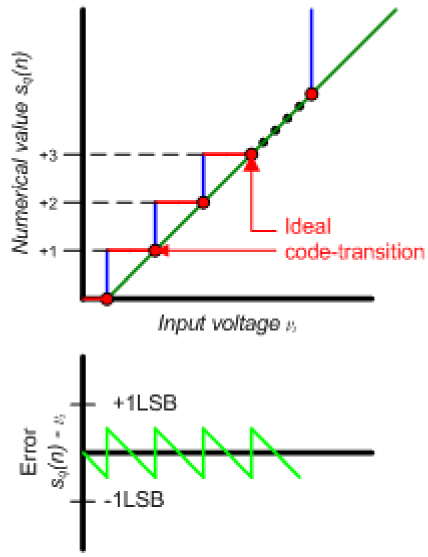


Figure 2.8: Ideal transitions in an ADC that uses rounding.

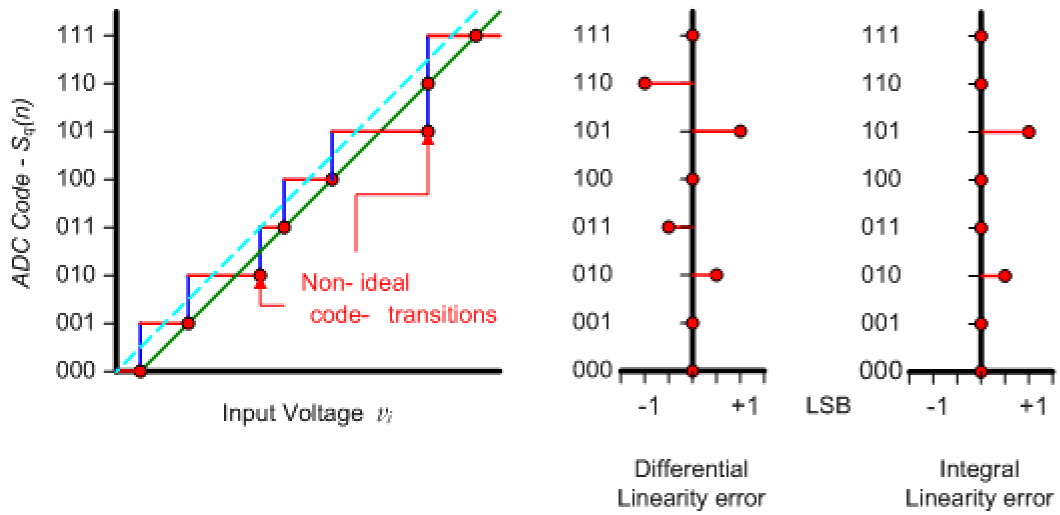


Figure 2.9: Non-ideal code and differential and integral linearity errors

The following observations can be made from Figure 2.9.

- Integral linearity error is equal to the sum of the differential linearity errors from 000.
- The code 101 will never be generated because the output switches from 101 to 111 directly. This code is called a missing code

If the differential linearity errors are less than one LSB, there are no missing codes. A good ADC should have a monotonic input-output relationship and there should be no missing codes. The slope of the input out relationship is called the gain of the ADC and should be unity.

The basic building block of an ADC is a logic gate shown in Figure 2.10. The input voltage is compared with a reference voltage. The output of the logic gate is “Low” (essentially zero) if the input voltage is less than the reference voltage but is “High” if the input voltage is greater than the reference voltage.

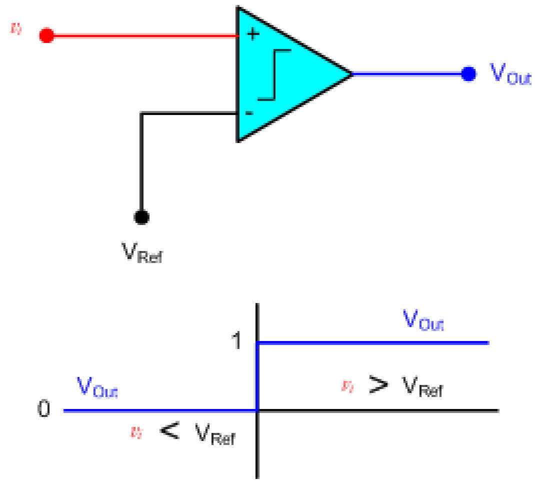


Figure 2.10: Logic gate used as a building block of an ADC

B. Successive Approximation converter

These converters use the well known successive approximation method. The process is implemented using a circuit whose basic form is shown in Figure 2.11. Consider a three-bit converter that has eight discreet levels. Also consider that the quantized equivalent of the input signal is greater than 101 (binary) but less than 110 (binary). The signal is applied to the logic gate as shown in the figure.

1. When a "Start" is received from a micro- processor, the ADC logic is reset. The SA Logic generates a number equal to half the largest number it can handle (it is binary 100 in this case). This signal is applied to a D/A converter that provides an analog voltage equivalent to the input binary value. The output of the D/A converter is applied to the logic gate. Because the input signal is larger than 0.5 times V_{Fs}, the output of the logic gate is "1".

2. The output of the logic gate is applied to the SA Logic. At the next clock pulse, an output code "1" is generated and saved as the high end bit of the conversion process. The binary output of the SA Logic is also increased to 0.75 times the largest number it can handle (it is binary 110 in this case). This binary output is applied to the D/A converter that generates a voltage equivalent to the input binary number and applies it to the logic gate. Because this voltage is larger than the input signal, the output of the logic gate is "0" which is received by the SA Logic.

3. The output of the logic gate is applied to the SA Logic. At the next clock pulse, an output code "0" is generated and saved as the second highest bit of the conversion process. The binary output of the SA Logic is reduced to half way between the previous two outputs (it is 101 in this case). The output of the SA Logic is applied to the D/A converter that provides an equivalent analog output to the Logic gate. Because the signal is larger than the output of the D/A converter, the output of the Logic gate generate is "1".

4. The output of the logic gate is applied to the SA Logic. At the next pulse, the SA Logic generates the LSB of the code as "1". The converted binary number is now 101 (101).

An eight-bit ADC has 256 discreet levels. If a signal greater than that equivalent to 145 binary and smaller than that equivalent to 146 binary is applied the outputs identified in Table 2.1 are generated.

Table 2.1: Outputs Generated by and 8-bit ADC

Clock cycle	Bit #	H/L	V_{DAC}	Output of Logic gate	ΔV_{DAC}
1			128	H	+64
2	7	1	192	L	-32
3	6	0	160	L	-16
4	5	0	144	H	+8
5	4	1	152	L	-4
6	3	0	148	L	-2
7	2	0	146	L	-1
8	1	0	145	H	0
9	0	1			

The output of the ADC is 10010001 ($2^7+2^4+2^0=128+16+1=145$)

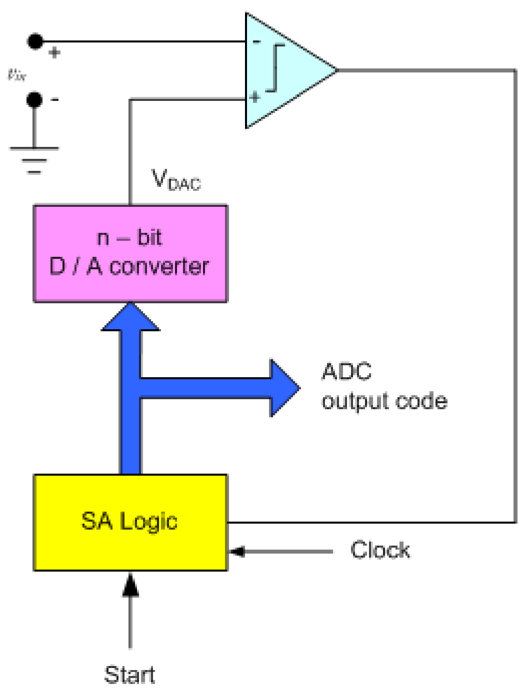


Figure 2.11: Basic circuit for successive approximation ADC

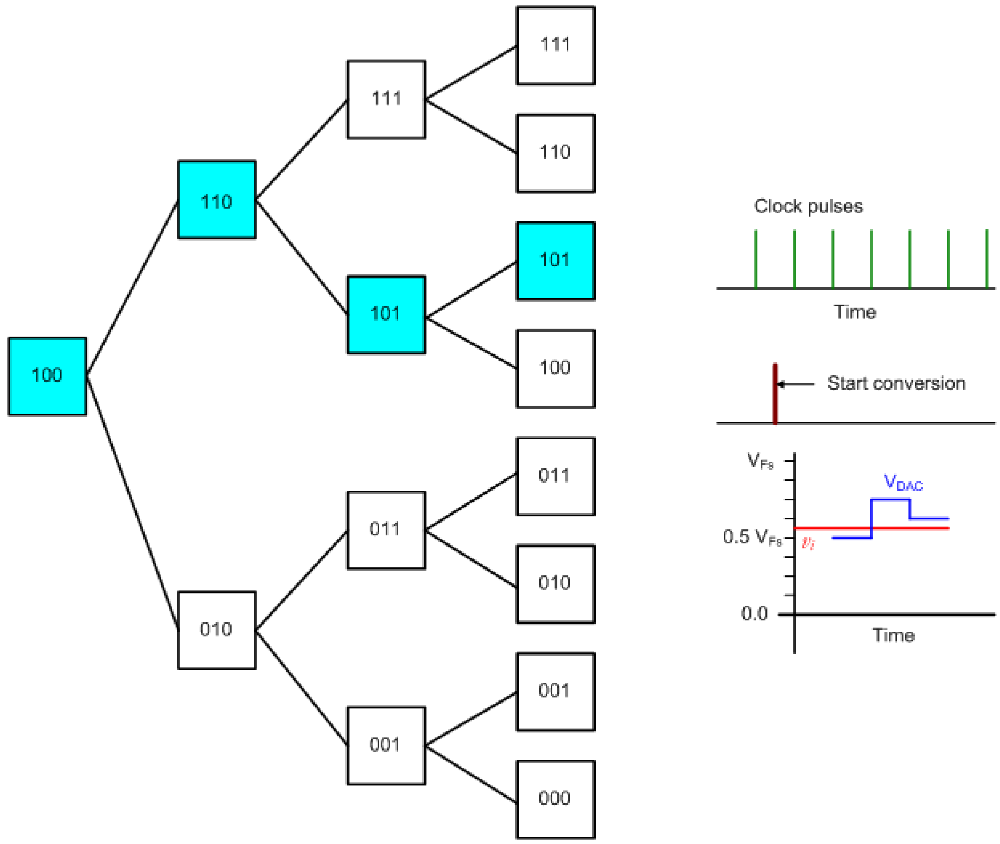


Figure 2.12: Successive approximation sequence and timing diagram

C. Flash converter

Some high-end relays use flash converters. At this time, flash converters of 12 or fewer bits are commercially available. The arrangement of a 3-bit flash converter is shown in Figure 2.13. The maximum voltage handled by the converter is applied to a voltage divider. Some designs use resistance dividers while the others use capacitive voltage dividers. The hardware is arranged to provide outputs in which the numbers are rounded instead of truncated. Because no switching or iterative approximations are used in these devices, these A/D converters are faster than the other types of converters.

The input signal is compared with the different levels of the voltage divider by individual logic gates. The output of the logic gates connected to the divider voltages that are less than the input voltage becomes high. This information is converted to the binary format by using a combinational logic circuit.

If more bits are needed, pipe line architecture flash converter can be selected.

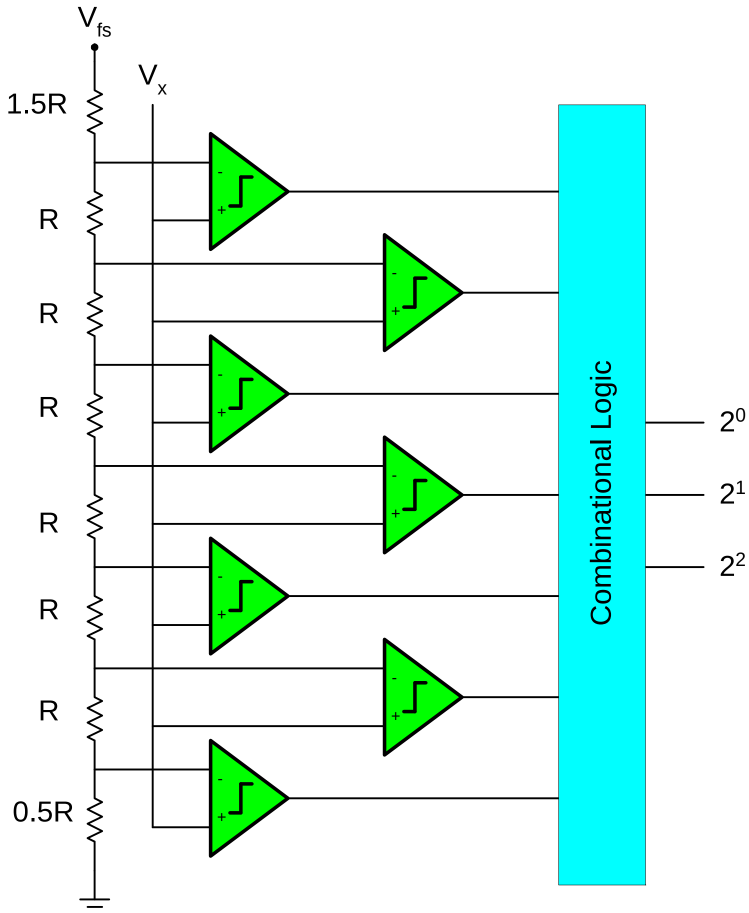


Figure 2.13: Circuit arrangement of a three-bit flash converter

3 PHASOR ESTIMATION TECHNIQUES

Several phasor estimation techniques have been proposed over the years. These techniques can be classified in the following categories.

- Discrete Fourier Transform
- Cosine algorithm
- Least squares algorithm
- Kalman filtering
- Wavelet transform

In this section, Discrete Fourier Transform, Cosine algorithm and least squares techniques are described. Most of the relays marketed today use one or a combination of two of these techniques.

3.1 Discrete Fourier Transform Technique

The Discrete Fourier Transform (DFT) technique is a short-time variation of the Fourier analysis. Like the Fourier analysis, the DFT assumes that a signal is made up of a fundamental frequency and harmonics of that frequency. While the Fourier transform is applied to signals in the continuous time domain, the DFT is applied to time-domain signals represented by sequences of numbers. Another major difference is that in the Fourier transform, the signal is assumed to exist from time $-\infty$ to $+\infty$ but in the DFT, the signal exists for a small duration of time (called window). The components of different frequencies determined by the DFT analysis can be combined to recreate the original waveform. The following example demonstrates this feature.

Figure 3.1 shows one cycle of a waveform that consists of a fundamental frequency component and a 3rd harmonic component. This waveform can be split into two sinusoids that, when combined together, recreate the original waveform. Figure 3.2 shows the fundamental frequency and third harmonic components of the waveform shown in Figure 3.1. The analysis provides the peak value and phase angle of the components. In this case, the phase angle of the third harmonic, referenced to the fundamental component, is 45^{o}. The components of other harmonic frequencies determined by the transform are zero. Combining the two components, shown in Figure 3.2, at each instant of time provide the original waveform shown in Figure 3.1.

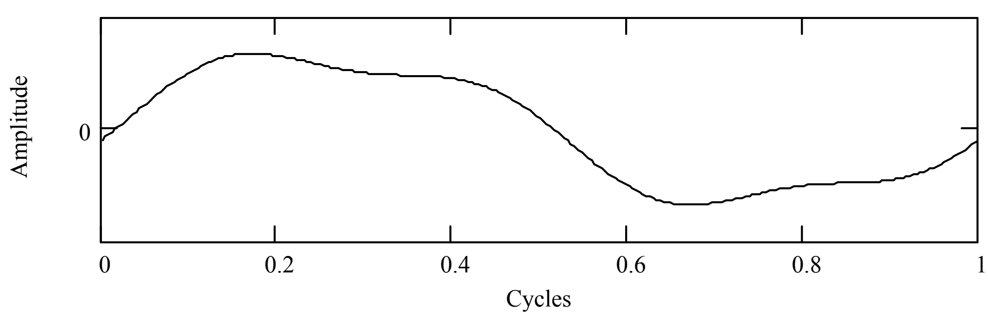


Figure 3.1: A waveform consisting of a fundamental and a 3rd harmonic component

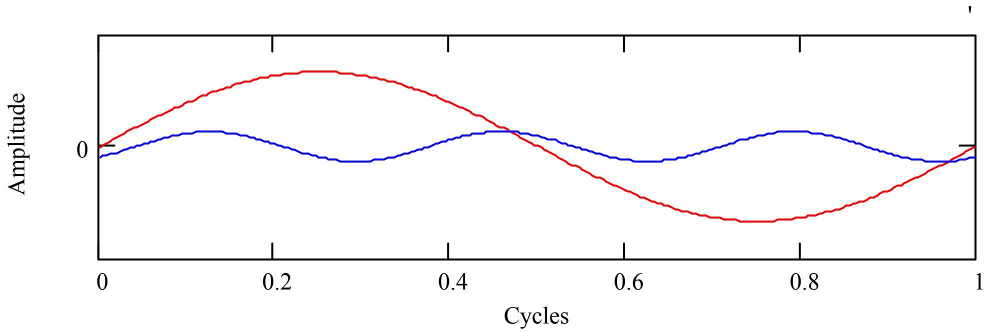


Figure 3.2: Sinusoidal components of the waveform of Figure 3.1

The implementation of the following equation performs the Discrete Fourier Transform of an input waveform.

$$V_h = \frac{2}{N}\sum_{n=0}^{N-1} v_n e^{-jn\frac{2\pi h}{N}} \tag{3.1}$$

In this equation,

- v is the instantaneous value of the voltage,
- n is the nth sample in the data window,
- V is the phasor of the voltage,
- h is the order of the harmonic, and
- N is the number of samples in a data window

When h = 0, the equation calculates the DC component of the waveform, when h = 1 the equation calculates the fundamental frequency component and so on.

Now, substituting the exponential term with its sinusoidal equivalent provides the following equation.

$$V_h = \frac{2}{N}\sum_{n=0}^{N-1} v_n \left[\cos\left(\frac{2\pi nh}{N}\right) - j\sin\left(\frac{2\pi nh}{N}\right)\right] \tag{3.2}$$

The real and imaginary parts of the phasor are given by the following equations.

$$V_h \cos\theta = \frac{2}{N}\sum_{n=0}^{N-1} v_n \cos\left(\frac{2\pi nh}{N}\right) \tag{3.3}$$

$$V_h \sin\theta = \frac{2}{N}\sum_{n=0}^{N-1} v_n \sin\left(\frac{2\pi nh}{N}\right) \tag{3.4}$$

In this equation, θ is the angle of the phasor. The application of Equation 3.3 and 3.4 consists of two steps. The first step is to obtain a weighted value of each sample by multiplying the value of the sample with a weight that is given by the terms of the sinusoids corresponding to the position of the sample in the data window. The second step is to add the weighted values of the N

weighted samples (in the data window) to obtain the real and the imaginary parts of the phasors. The peak value and phase angle of the phasor can be calculated from these components.

3.1.1 Selecting the data window

A review of Equations 3.1 to 3.4 reveals that *N* samples, from a data window, are used for computing the phasors of the fundamental and harmonic frequencies. The size of the data window, used in an application, has an impact on the results. It is, therefore, an important issue that needs some discussion.

If a signal consists of the fundamental frequency only, the data window should be one period of the waveform. The window could also be one, two or n multiples (n is an integer) of the time period of the waveform.

Now consider a case in which the waveform consists of the fundamental and harmonic frequencies. The window in this case must be the time period of the fundamental frequency. This window is used for calculating the phasors of the fundamental. For calculating the phasors of the harmonic frequencies, the time period of the fundamental frequency should be used. In other words, the time period of the component of the lowest frequency must be used for calculating phasors by the DFT technique.

3.1.2 An example

An example is included in this section to explain the procedure used to calculate the phasors by the application of DFT technique. Consider that the current displayed in Figure 3.1 is sampled at 480 Hz (eight times a period of the fundamental frequency). The quantized values of the samples are listed in Column 2 of Table 3.1.

Table 3.1: Quantized values of the samples, weights and weighted samples

Sample No.	Quantized Value	Weight $\cos(\Delta\theta)$	Weight $\sin(\Delta\theta)$	Weighted sum Real	Weighted sum Imaginary
0	-14	1.000	0.000	-14	0
1	91	0.707	-0.707	64	-64
2	86	0.000	-1.000	0	-86
3	71	-0.707	-0.707	-50	-50
4	14	-1.000	0.000	-14	0
5	-91	-0.707	0.707	64	-64
6	-86	0.000	1.000	0	-86
7	-71	0.707	0.707	-50	-50
			Total	0	-400

The fundamental frequency phasor is one quarter (2/8) of the sum calculated in Columns 5 and 6 of Table 3.1. The phasor, therefore, is 0 – j 100.

For calculating the phasors of the harmonic components, weights calculated at those frequencies must be used. The weights corresponding to the second, third and fourth harmonic frequencies are given in Table 3.2. The weights corresponding to other higher order harmonics can be similarly defined. The use of the quantized values of the signal and the weights corresponding to a harmonic frequency provides the phasor representing the component of that frequency. Equation 3.5 lists the computed values of the phasors by using the Discrete Fourier Transform. The first column lists the order of the harmonic whose phasor is listed in that row. The second column lists the phasor in the rectangular form. The third column lists the peak value of the phasor and the fourth column lists the phase angle of the phasor.

The first row of Equation 3.5 lists the DC component of the waveform. The result is zero; it makes sense because there is no dc component in this waveform. The second row lists the fundamental frequency component of the waveform. This component has amplitude of 100 with a phase angle of 270°. The third row lists the second harmonic component. This component is not present in the waveform and, therefore, its value is zero. Similarly, the fourth row lists the third harmonic component. This phasor has a peak value of 20 and a phase angle of 225°.

Table 3.2: Weights for calculating the real and imaginary components of harmonics

Second Harmonic		Third Harmonic		Fourth Harmonic	
Weights for Real part	Weights for Imaginary Part	Weights for Real part	Weights for Imaginary Part	Weights for Real part	Weights for Imaginary Part
1	0	1.000	0.000	1.000	0.000
0	-1	-0.707	-0.707	-1.000	0.000
-1	0	0.000	1.000	1.000	0.000
0	1	0.707	-0.707	-1.000	0.000
1	0	-1.000	0.000	1.000	0.000
0	-1	0.707	0.707	-1.000	0.000
-1	0	0.000	-1.000	1.000	0.000
0	1	-0.707	0.707	-1.000	0.000

$$\mathrm{V} = \begin{bmatrix} 0 & 0 & 0 & \\ 1 & 0 - j100 & 100 & 270^{\circ} \\ 2 & 0 - j0 & 0 & 0^{\circ} \\ 3 & -14 - j14 & 20 & 225^{\circ} \\ 4 & 0 + j0 & 0 & 0^{\circ} \end{bmatrix} \tag{3.5}$$

The equation that will recreate the original waveform is

$$v_{rec}\left(t\right) = 100\cos\left(2\pi\, ft + 270^{\circ}\right) + 20\cos\left(6\pi\, ft + 225^{\circ}\right) \tag{3.6}$$

If the dc term were not zero, the magnitude for this term should be corrected by dividing by two. The magnitude of the (N/2)th harmonic is not computed consistently. The result depends on the orientation of the samples with respect to the zero crossing of that component.

3.1.3 Impact of components of frequencies larger than the Nyquist frequency

If frequencies above the Nyquist frequency, (N/2)th harmonic frequency were present in the waveform, aliasing would have occurred. The energy from those frequencies would have fallen into the terms from the dc to the (N/2)th harmonic. It would have corrupted the estimates of those frequencies. The component of the (1+N/2)th harmonic would have shown up as a component of (-1+N/2)th harmonic. A component of (2+N/2)th harmonic would have shown up as a component of the (-2+N/2)th harmonic. This pattern continues between the Nyquist frequency and dc as the aliasing frequency increases to infinity.

To illustrate this phenomenon, consider that the waveform contains two components, a fundamental frequency component and the fifth harmonic components. The waveform is now represented by the following equation.

$$v(t) = 100\cos\left(2\pi ft - 90^{o}\right) + 20\cos\left(10\pi ft - 45^{o}\right) \tag{3.7}$$

Sampling this waveform eight times per period and calculating the components of different frequencies by the DFT technique provides the following estimates

$$V = \begin{bmatrix} 0 & 0 & 0 & \\ 1 & 0 - j100 & 100 & 270^{o} \\ 2 & 0 - j0 & 0 & 0^{o} \\ 3 & -14 + j14 & 20 & 135^{o} \\ 4 & 0 + j0 & 0 & 0^{o} \end{bmatrix} \tag{3.8}$$

There is no third harmonic component present in the waveform but the DFT analysis shows that it is present. The Nyquist frequency is the 4^{th} harmonic and, the energy of the 5th harmonic component aliases in to the 3rd harmonic [-(5-4)+4].

Now consider that the voltage waveform is made up of the fundamental and the seventh harmonic expressed by the following equation.

$$v(t) = 100\cos\left(2\pi ft - 90^{o}\right) + 20\cos\left(14\pi ft - 45^{o}\right) \tag{3.9}$$

Sampling this waveform eight times per period and calculating the components of different frequencies by the DFT technique provides the following estimates.

$$V = \begin{bmatrix} 0 & 0 & 0 & \\ 1 & -14 - j86 & 87 & 261^{o} \\ 2 & 0 - j0 & 0 & 0^{o} \\ 3 & 0 + j0 & 0 & 0^{o} \\ 4 & 0 + j0 & 0 & 0^{o} \end{bmatrix} \tag{3.10}$$

Recollect that the Nyquist frequency is the 4th harmonic and, therefore, the energy of the 7th harmonic component aliases in to the fundamental frequency [-(7-4)+4]. The fundamental frequency estimate has, therefore, been corrupted in magnitude as well as in phase.

Now consider that the voltage waveform is made up of the fundamental and the eight harmonic expressed by the following equation.

$$v(t) = 100\cos\left(2\pi ft - 90^{o}\right) + 20\cos\left(16\pi ft - 45^{o}\right) \quad (3.11)$$

Sampling this waveform eight times per period and calculating the components of different frequencies by the DFT technique provides the following estimates.

$$\mathrm{V} = \begin{bmatrix} 0 & 28 & 0 & \\ 1 & 0-j100 & 100 & 270^{o} \\ 2 & 0-j0 & 0 & 0^{o} \\ 3 & 0+j0 & 0 & 0^{o} \\ 4 & 0+j0 & 0 & 0^{o} \end{bmatrix} \quad (3.12)$$

Recollect that the Nyquist frequency is the 4th harmonic and, therefore, the energy of the 8th harmonic component aliases into dc [-(8-4)+4]. Because of this aliasing process, the eight-harmonic component appears to be a dc component even when there is no dc component in the voltage waveform.

The energy for the 10th harmonic would fall into the 2nd harmonic bin and this pattern would continues to the harmonic of infinite order.

This example demonstrates the need of pre-filtering for removing the energies in the components of frequencies above the Nyquist frequency.

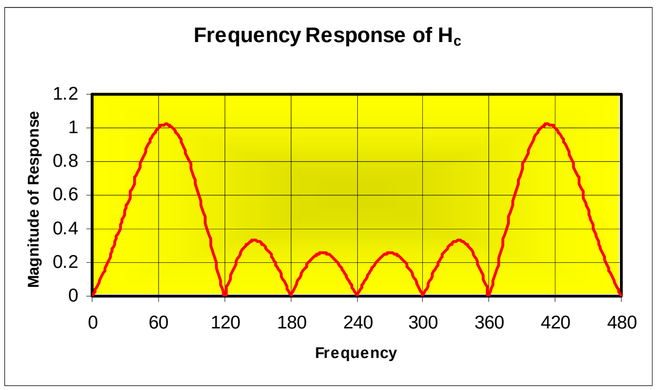
In many applications, estimates of the fundamental frequency are needed. The DFT approach is used to obtain these estimates. In other applications, the estimates of a spectrum of frequencies are needed. An example of such an application is the power quality analysis. For such applications, the efficiency of the estimation process is much more if the data is sampled at a rate based on an even power of 2 (8, 16, 32, 64 etc.) and the Fast Fourier Transform is used.

The benefit of the DFT technique is that the fundamental frequency component of a fault voltage or current can be obtained. A relay can be set to operate on this quantity ignoring all other components in the waveform. In the case of a transformer differential relay, the DFT can be used to compute the fundamental frequency (50 or 60Hz) component and the 2nd and 5th harmonics. The level of 2nd harmonic can be used to restrain from operating on inrush and the level of 5th harmonic can be used to restrain during over-excitation.

3.2 Cosine Filter

Figure 3.3 shows the frequency response of an eight-sample DFT algorithm. The frequency response H_c is referred to as the response of the Cosine filter and the response H_s is referred to as the response of the Sine filter. The frequency response of Sine and Cosine filters is discussed in detail in section 3.4. It is interesting to note that the Sine filter suppresses the high frequency components more than the suppression provided by the Cosine filter. An evaluation of the response of the Sine and Cosine filters to the exponentially decaying dc reveals that the Cosine filters suppress the decaying dc components more than the Sine filters do.

Consider that phasors are being processed eight times a cycle of the fundamental frequency by a Cosine filter. The output at an instant t_0 may be identified as the real part of the phasor and the output of the filter at time t_2 would be the imaginary part of the phasor if ωt_2 is equal to $\omega t_1+0.5\pi$. The imaginary part, therefore, is calculated from the sample occurring one-quarter cycle later than the real part in the output samples. This delays the phasor estimating in Cosine filter by one-quarter cycle compared with the DFT filter. However, the phasors calculated in this manner are affected to a lesser degree by the presence of decaying dc components in voltages and currents as compared to the effect of the dc decaying component on the phasors calculated by the DFT. This makes the cosine filter faster in calculating acceptable estimates in some cases than the DFT filter.



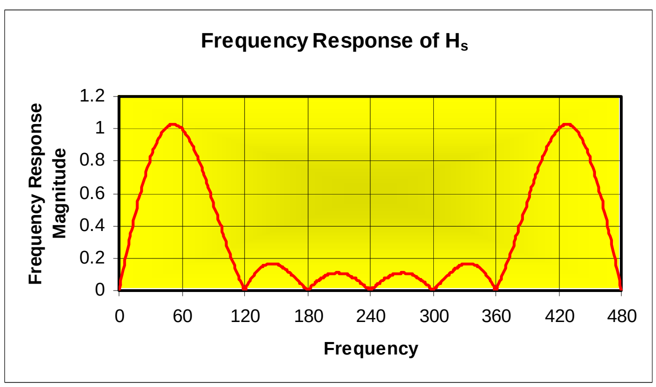


Figure 3.3: Frequency response of an 8-point DFT (a) cosine filter and (b) sine filter

3.3 Least Squares Technique

Least Error Squares (LES) technique [1] is used to estimate the phasors of the fundamental and harmonic frequency components of voltages and currents. It is based on minimizing the mean-square error between the actual and assumed waveforms. The voltage and/or current waveform is modeled as a combination of the fundamental frequency component, an exponentially decaying dc component and harmonics of specified orders.

$$v(t) = V_0 e^{-\frac{t}{\tau}} + \sum_{n=1}^{N} V_n \sin(n\omega_0 t + \theta_n) \tag{3.13}$$

where: $v(t)$ is the instantaneous value of the voltage at time t .
τ is the time constant of the decaying dc component.
N is the highest order of the harmonic component present in the signal.
ω_0 is the fundamental frequency of the system.
V_0 is the magnitude of the dc offset at $t = 0$.
V_n is the peak value of the n^{th} harmonic component.
θ_n is the phase angle of the n^{th} harmonic component.

Expressing the decaying dc component by the Taylor series expansion and retaining the first two terms, the following equation is obtained.

$$v(t) = V_0 - \left(\frac{V_0}{\tau}\right)t + \sum_{n=1}^{N} V_n \sin(n\omega_0 t + \theta_n) \tag{3.14}$$

Assume that the voltage is composed of an exponentially decaying dc component, the fundamental frequency component and components of the second, third, fourth and fifth harmonics. For $t = t_1$, Equation 3.14 can be, therefore, expressed as

$$\begin{aligned} v(t_1) = {} & V_0 - \left(\frac{V_0}{\tau}\right)t_1 + V_1 \sin(\omega_0 t_1 + \theta_1) + V_2 \sin(2\omega_0 t_1 + \theta_2) \\ & + V_3 \sin(3\omega_0 t_1 + \theta_3) + V_4 \sin(4\omega_0 t_1 + \theta_4) + V_5 \sin(5\omega_0 t_1 + \theta_5) \end{aligned} \tag{3.15}$$

Using trigonometric identities, the sinusoids in Equation 3.15 can be expanded. The right-hand of the resulting equation contains the real and imaginary components of the phasors of each frequency and coefficients that are sin and cosines of $\omega_0 t_1$.

Considering that the signal contains N harmonic components. Also consider that the signal is sampled at intervals of Δt seconds which corresponds to taking P samples in cycle of the fundamental frequency. Each sample can be expressed as a function of the phasors and coefficients. If [(2N+2)+1] samples are expressed by such equations, the following matrix equation is obtained.

$$\underset{(2N+3)\times(2N+2)}{[A]} \; \underset{(2N+2)\times 1}{[X]} = \underset{(2N+3)\times 1}{[v]} \tag{3.16}$$

The least error squares estimate of $[X]$ is given by the following equation.

$$\begin{aligned} [X] &= \left[[A]^T [A]\right]^{-1} [A]^T [v] \\ &= [A]^{\dagger} [v] \end{aligned} \tag{3.17}$$

In this equation $[A]^{\dagger}$ is the left pseudo-inverse of $[A]$. Details of the derivation of the least squares approach is given in Appendix A.

The elements of the 3rd and 4th rows of $[A]^{+}$, are the coefficients of the filter for estimating the real and imaginary components of the fundamental frequency phasor of the signal. These coefficients can be calculated in the off-line mode.

Consider an example using a sampling frequency of 720 Hz (P=12). The time coinciding with the seventh sample is considered to be zero. The filter coefficients, for estimating the real and imaginary components of the fundamental frequency phasor of the signal, are given in Table 3.4. The real and imaginary components of the fundamental frequency phasor are calculated by multiplying the coefficients with the samples. The transfer function of the cosine and sine filters in the z -plane is given by

$$\begin{aligned} &C(1)z^{+6} + C(2)z^{+5} + C(3)z^{+4} + C(4)z^{+3} + C(5)z^{+2} + C(6)z^{+1} + C(7)z^{0} \\ &+C(8)z^{-1} + C(9)z^{-2} + C(10)z^{-3} + C(11)z^{-4} + C(12)z^{-5} + C(13)z^{-6} \end{aligned} \tag{3.18}$$

The magnitude response of the LES technique can be obtained by using the numerical values of the filter coefficients and substituting z with $e^{j\omega\Delta t}$ in Equation 3.22 and evaluating the resulting equation. The responses are shown in Figure 3.4.

Table 3.4: Filter coefficients for the Least Error Squares technique for 720 Hz. sampling frequency for 60 Hz signals

Coefficient Number	Cosine Filter (Real)	Sine Filter (Imaginary)
C(1)	0.31100423	-0.08695652
C(2)	-0.08333333	-0.13709119
C(3)	-0.14433757	-0.09057971
C(4)	-0.16666667	0.00724638
C(5)	-0.14433757	0.07608695
C(6)	-0.08333333	0.15158394
C(7)	0.00000000	0.15942029
C(8)	0.08333333	0.15158394
C(9)	0.14433757	0.07608695
C(10)	0.16666667	0.00724638
C(11)	0.14433757	-0.09057971
C(12)	0.08333333	-0.13709119
C(13)	-0.31100423	-0.08695652

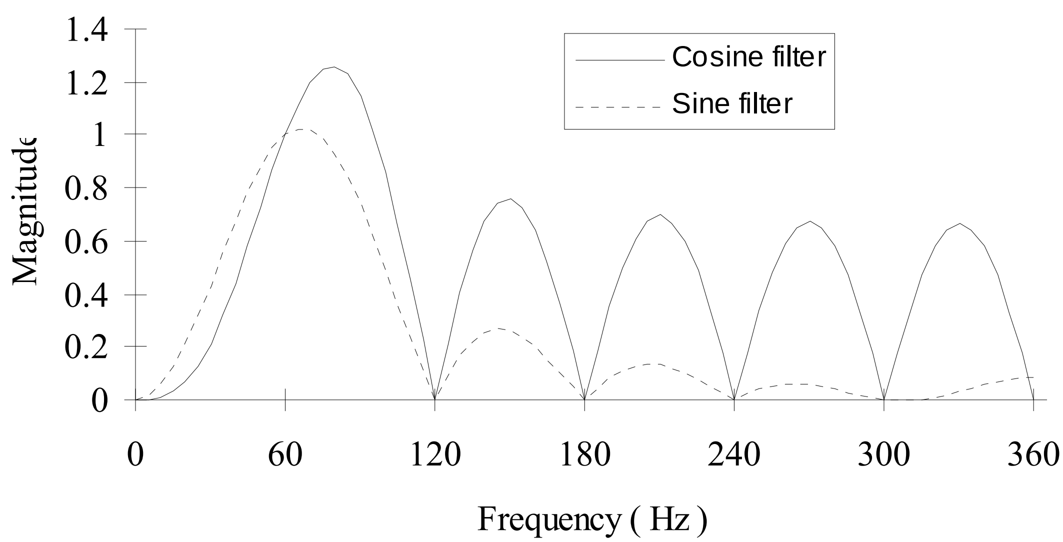


Figure 3.4: Magnitude response of the Least Error Squares technique

3.4 Frequency Response of Algorithms

The major purpose of an algorithm is to calculate the magnitudes and phase angles of phasors. While the algorithms perform this function, they also have a natural property of filtering. This property can be determined by mathematical analysis, which is demonstrated in this section.

Consider the DFT process, which is used for calculating the phasor for representing the fundamental frequency component of the waveform shown in Figure 3.1. Also consider that the fundamental frequency is 60 Hz. The time between two consecutive samples and the angle at the fundamental frequency it represents are given in Equation 3.23. The weights used for calculating the phasors are given in Table 3.1 and are reproduced in Table 3.3. These weights, represented as a function of time are shown in Figure 3.5. It is assumed in this case that the time is zero at the latest sample of the data window.

$$\Delta T = \frac{1}{480} \quad s$$
$$\Delta\theta = \frac{2\pi}{8} \quad r \qquad (3.19)$$

Table 3.3: Weights for calculating phasors by using an 8-sample DFT

Sample No.	Weight cos(Δθ)	Weight sin(Δθ)
0	1.000	0.000
1	0.707	-0.707
2	0.000	-1.000
3	-0.707	-0.707
4	-1.000	0.000
5	-0.707	0.707
6	0.000	1.000
7	0.707	0.707

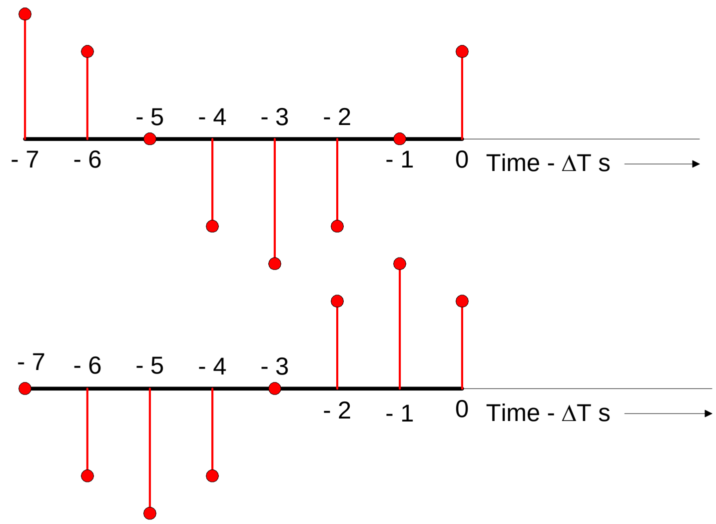


Figure 3.5: Weights used by the 8-samples DFT as a function of time

The transfer functions of these processes can be expressed in the z-transform domain as follows.

$$H_c(\omega) = \frac{1}{4}\left[1.0z^7 + \frac{1}{\sqrt{2}}z^6 + 0.0z^5 - \frac{1}{\sqrt{2}}z^4 - 1.0z^3 - \frac{1}{\sqrt{2}}z^2 + 0.0z^1 + \frac{1}{\sqrt{2}}z^0\right]$$
$$H_s(\omega) = \frac{1}{4}\left[0.0z^7 - \frac{1}{\sqrt{2}}z^6 - 1.0z^5 - \frac{1}{\sqrt{2}}z^4 + 0.0z^3 + \frac{1}{\sqrt{2}}z^2 + 1.0z^1 + \frac{1}{\sqrt{2}}z^0\right] \quad (3.20)$$

The frequency-responses in the frequency domain are obtained by replacing z by $e^{j\omega\Delta T}$; these responses are as follows.

$$H_c(\omega) = \frac{1}{4}\left[1.0e^{j7\omega\Delta T} + \frac{1}{\sqrt{2}}e^{j6\omega\Delta T} - \frac{1}{\sqrt{2}}e^{j4\omega\Delta T} - 1.0e^{j3\omega\Delta T} - \frac{1}{\sqrt{2}}e^{j2\omega\Delta T} + \frac{1}{\sqrt{2}}\right]$$
$$H_s(\omega) = \frac{1}{4}\left[-\frac{1}{\sqrt{2}}e^{j6\omega\Delta T} - 1.0e^{j5\omega\Delta T} - \frac{1}{\sqrt{2}}e^{j4\omega\Delta T} + \frac{1}{\sqrt{2}}e^{j2\omega\Delta T} + 1.0e^{j\omega\Delta T} + \frac{1}{\sqrt{2}}\right] \quad (3.21)$$

The terms whose coefficients are zero have been omitted and the term e^{j0} has been replaced by one in these equations. Substituting a selected frequency in the transfer functions and evaluating them provides the response of the functions at that frequency. The result is a complex number; the magnitude and phase angle of the response is calculated from the complex number.

Consider that the selected frequency is 60 Hz; ΔT at this frequency is $\frac{120\pi}{480}$ $(=\frac{\pi}{4})$. Equation 3.26 is obtained by substituting the value of ω and ΔT in Equation 3.25.

$$H_c(\omega)=\frac{1}{4}\left[1.0e^{j\frac{7\pi}{4}}+\frac{1}{\sqrt{2}}e^{j\frac{6\pi}{4}}-\frac{1}{\sqrt{2}}e^{j\frac{4\pi}{4}}-1.0e^{j\frac{3\pi}{4}}-\frac{1}{\sqrt{2}}e^{j\frac{2\pi}{4}}+\frac{1}{\sqrt{2}}\right]$$
$$H_s(\omega)=\frac{1}{4}\left[-\frac{1}{\sqrt{2}}e^{j\frac{6\pi}{4}}-1.0e^{j\frac{5\pi}{4}}-\frac{1}{\sqrt{2}}e^{j\frac{4\pi}{4}}+\frac{1}{\sqrt{2}}e^{j\frac{2\pi}{4}}+1.0e^{j\frac{\pi}{4}}+\frac{1}{\sqrt{2}}\right] \tag{3.22}$$

Replacing the exponential terms with their cosine and sine functions provides Equation 3.27.

$$\begin{aligned}H_c(\omega)=&\frac{1}{4}\left[1.0\cos\left(\tfrac{7\pi}{4}\right)+\frac{1}{\sqrt{2}}\cos\left(\tfrac{6\pi}{4}\right)-\frac{1}{\sqrt{2}}\cos\left(\tfrac{4\pi}{4}\right)-1.0\cos\left(\tfrac{3\pi}{4}\right)-\frac{1}{\sqrt{2}}\cos\left(\tfrac{2\pi}{4}\right)+\frac{1}{\sqrt{2}}\right]\\&-j\frac{1}{4}\left[1.0\sin\left(\tfrac{7\pi}{4}\right)+\frac{1}{\sqrt{2}}\sin\left(\tfrac{6\pi}{4}\right)-\frac{1}{\sqrt{2}}\sin\left(\tfrac{4\pi}{4}\right)-1.0\sin\left(\tfrac{3\pi}{4}\right)-\frac{1}{\sqrt{2}}\sin\left(\tfrac{2\pi}{4}\right)\right]\\H_s(\omega)=&\frac{1}{4}\left[-\frac{1}{\sqrt{2}}\cos\left(\tfrac{6\pi}{4}\right)-1.0\cos\left(\tfrac{5\pi}{4}\right)-\frac{1}{\sqrt{2}}\cos\left(\tfrac{4\pi}{4}\right)+\frac{1}{\sqrt{2}}\cos\left(\tfrac{2\pi}{4}\right)+1.0\cos\left(\tfrac{\pi}{4}\right)+\frac{1}{\sqrt{2}}\right]\\&-j\frac{1}{4}\left[-\frac{1}{\sqrt{2}}\sin\left(\tfrac{6\pi}{4}\right)-1.0\sin\left(\tfrac{5\pi}{4}\right)-\frac{1}{\sqrt{2}}\sin\left(\tfrac{4\pi}{4}\right)+\frac{1}{\sqrt{2}}\sin\left(\tfrac{2\pi}{4}\right)+1.0\sin\left(\tfrac{\pi}{4}\right)\right]\end{aligned} \tag{3.23}$$

Substituting the values of the trigonometric functions in this equation provides Equation 3.28.

$$\begin{aligned}H_c(120\pi)=&\frac{1}{4}\left[\frac{1}{\sqrt{2}}+0.0+\frac{1}{\sqrt{2}}+\frac{1}{\sqrt{2}}-0.0+\frac{1}{\sqrt{2}}\right]\\&-j\frac{1}{4}\left[-\frac{1}{\sqrt{2}}-\frac{1}{\sqrt{2}}-0.0-\frac{1}{\sqrt{2}}-\frac{1}{\sqrt{2}}\right]\\H_s(120\pi)=&\frac{1}{4}\left[-0.0+\frac{1}{\sqrt{2}}+\frac{1}{\sqrt{2}}+0.0+\frac{1}{\sqrt{2}}+\frac{1}{\sqrt{2}}\right]\\&-j\frac{1}{4}\left[\frac{1}{\sqrt{2}}+\frac{1}{\sqrt{2}}-0.0+\frac{1}{\sqrt{2}}+\frac{1}{\sqrt{2}}\right]\end{aligned} \tag{3.24}$$

The frequency responses are therefore

$$\begin{aligned}H_c(120\pi)&=\frac{1}{\sqrt{2}}+j\frac{1}{\sqrt{2}}\\H_s(120\pi)&=\frac{1}{\sqrt{2}}-j\frac{1}{\sqrt{2}}\end{aligned} \tag{3.25}$$

Notice that the magnitude of the frequency response of each process is 1.0 and the phase displacement between the two frequency responses is 90°. When the H_c function is used to process quantized data from a 60 Hz waveform, the output waveform will have the same magnitude as the input waveform but it will be displaced $+45^{\circ}$ with respect to the input. Similarly, when the H_s function is used to process quantized data from a 60 Hz waveform, the output will have the same magnitude as the input waveform but it will be displaced -45° with respect to the input.

The frequency responses of the 8-sample DFT algorithm to the signals of frequencies from 0 to 480 Hz are as shown in Figure 3.6. Three important observations from these responses are worth noticing. First, the frequency responses are zero at the second to sixth harmonic frequencies. Second, the frequency responses of H_c and H_s are 1.0 at 60 Hz (the fundamental frequency) but are different at other frequencies. The frequency responses are mirror images about the Nyquist frequency (one-half the sampling frequency = 240 Hz).

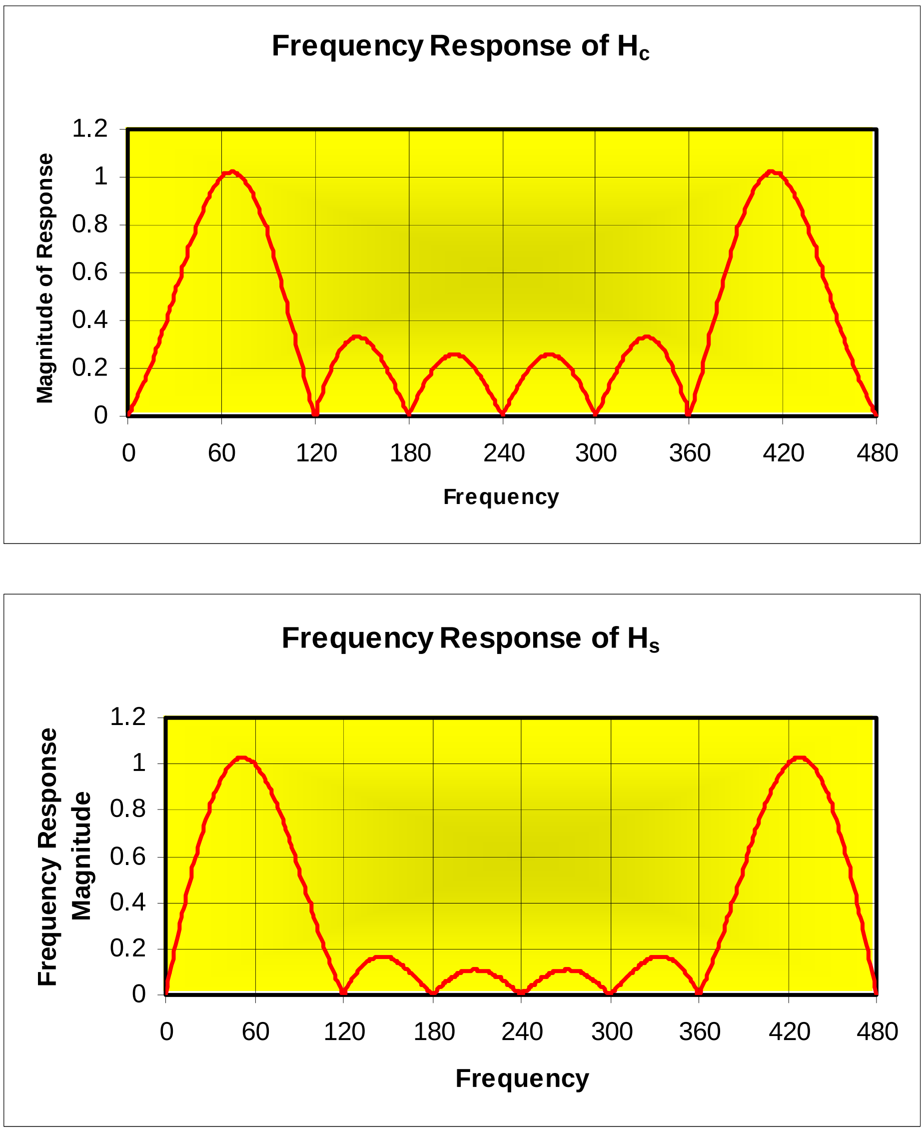


Figure 3.6: Frequency responses of the 8-sample DFT algorithm

3.5 Requirements and Phasor Estimation Properties of Filters used in Relays

This section evaluates some properties of digital filters when data is acquired on the occurrence of faults.

Faults in power systems are accompanied by noise in the form of harmonics, subsynchronous transients and an exponentially decaying dc offset. The levels of these components depend on the system parameters and the instant of fault occurrence. The filters used in relays must be able to reject all noise and components of non-fundamental frequencies so that the phasors of the fundamental frequency are accurately estimated. The filters should also have good transient response and a reasonable bandwidth.

Digital filters could be either infinite impulse response (IIR) or finite impulse response (FIR) type. The outputs of FIR filters depend on a finite-time-history of the input. On the other hand, the outputs of IIR filters depend on the history of all prior inputs. FIR filters subjectively make good sense for protection for two reasons:

- FIR filters quickly forget the prefault condition, and work on analyzing the data obtained during faults. Once the data-windows of the filters fill up with fault data, the estimated values of the voltage and current phasors are no longer corrupted with prefault data.
- FIR filters naturally have zeros in their frequency responses. It is relatively easy to put zeros where we want them, e.g., at dc and harmonics.

Other advantages of FIR filters are that they are 'linear-phase' filters, meaning they do not distort the phase of the input signal and they are simple to implement. All the phasor estimation techniques used in relays are implemented as FIR filters.

An issue with the digital filters discussed in this document is the removal of exponentially decaying dc offset found in fault currents immediately after the occurrence of a fault. The extent of this offset is determined by the instant of fault and the time constant of decay that is determined by the X/R ratio (reactance to resistance ratio) of the system. Since this ratio depends on the fault resistance, the time constant cannot be known beforehand**.**

The speed of fault-detection is also an important design aspect in a relay, since it reduces the total fault-clearing time. It is found that increasing the sampling rate marginally decreases the operating time of a relay at lower sampling rates, but at higher sampling rates, the advantage is minimal. In general, doubling the sampling rate from "*x*" to "2*x*" decreases the phasor estimation time by $1/(2x)$ cycle. Therefore, increasing the sampling rate from 4 to 8 decreases the phasor estimation time by one eighth of a cycle, but increasing it from 32 to 64 decreases the time only by about one quarter millisecond for 60 Hz phasors and 0.3 milliseconds for 50 Hz phasors. In addition, when the sampling rate is doubled, the calculations to be performed in the relay are approximately doubled. Increasing the speed by shortening the sampling window is another option. When data windows of one-half of one cycle of the frequency of interest are used, the filter looses the ability to reject even harmonics and dc component. Several approaches can be used to remove the decaying dc offset. One of the options is to use mimic filters during pre-processing of the inputs [21]. However, a mimic filter can remove the decaying dc component completely only if the time constant of the decaying dc is known before hand, which is usually not the case in a real power system. It is found that faster decaying rates are more detrimental to the accuracy of these filters [21]. The improvement in speed can also be achieved by decreasing the delay in the analog low-pass anti-aliasing filter shown in Figure 1.2.

4 FREQUENCY ESTIMATION

Microprocessor-based relays estimate power system frequency for several reasons. A few of the reasons are as follows.

- Power system frequency, along with other measurements, help in analyzing the performance of relays as well as the performance of power system during steady state and transient operations.
- Power system frequency, estimated by a relay, can be used (and is used in some relays) for controlling the rate of sampling voltages and currents to ensure accurate operation at all frequencies including the off-nominal frequencies.
- Frequency is used for a number of protection and control functions performed by relays. This includes under- and over-frequency, volts-per-hertz, load-shedding, synchrocheck, and automatic synchronizing functions. Often, rate-of-change of frequency is measured in addition to frequency. The accuracy with which these parameters should be measured depends on the accuracy and response time required by the application.

Frequency of a periodic waveform is a parameter that is not usually measured but is estimated from a system voltage or current. Distortions in these signals affect the accuracy of the estimates seriously. For example, if a simple zero-crossing detector is used for estimating frequency, a zero-crossing anomaly, when a current is used to measure frequency and current reversal occurs, could have a great impact on the accuracy of the measurement. When a voltage is used to measure frequency, a CVT-inducted transient could adversely impact the accuracy of the measured frequency. Because of these reasons, the algorithms for measuring frequency are designed to measure it from data of several periods of the signal.

There are, however, some applications that should respond to frequency changes in very short times. Examples of such applications include generator protection applied to small machines with low inertia, protection of variable frequency motors, or fast load-shedding schemes.

Some of the problems in measuring frequency by numerical algorithms include:

- Signal pollution, particularly with non-harmonic and non-periodic components that are encountered usually in the vicinity of non-linear loads.
- Signal distortions due to CVT transients, CT saturation and current reversals.
- Rapidly changing frequencies such as during emergency start-up of a small generator.

Typically, numerical algorithms for estimating frequency use sampled and quantized values of a selected voltage. A phase-to-phase voltage is preferred over a phase-to-ground voltage because the latter is usually more distorted. Some algorithms, however, use a composite signal, such as the positive-sequence voltage or another linear combination of the three phase voltages. In some applications, current waveforms are used for estimating frequency. These applications include differential relays, such as the line current differential or bus differential relays.

Out of the many techniques that are available for measuring frequency, zero crossing and phasor tracking techniques are described in this section. The issue of frequency tracking is then discussed.

4.1 Zero-Crossing Detection

The zero-crossing frequency estimation technique is a popular method that can be implemented either in the hardware or in the software of a microprocessor-based relay. The method measures the time between two zero crossings and calculates the frequency from that measurement. The time between two consecutive zero crossings is one-half the time of a period whereas the time between two alternate half cycles is the time of one period of the signal. Other variations of this approach measure the time between a selected number of zero crossings. These methods use the following generic equation.

$$f\left(t_M\right) = \frac{M-1}{2}\frac{1}{t_M - t_1} \tag{4.1}$$

where,

- M is the number of zero crossings
- t_M time of the mth zero crossing
- t_1 is the time of the first zero crossing
- f is the estimate of the frequency

Because this method is susceptible to spurious zero crossings, it is necessary that the input waveform be pre-filtered using either a low-pass or a band-pass filter. If implemented in software, the method requires the time of a zero crossing be estimated from the time stamps of the samples in its vicinity as shown in Figure 4.1 except when the samples are taken at the rate of tens of thousands per second.

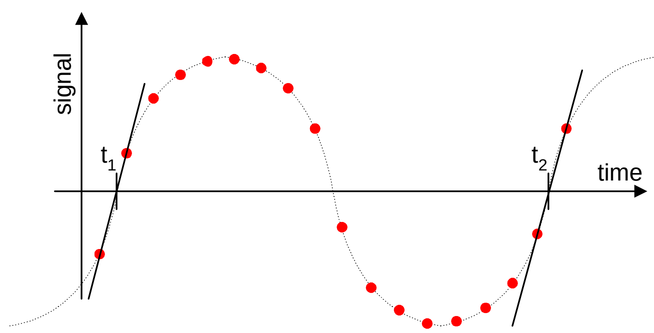


Figure 4.1: Software implementation of the zero-crossing detection

This method usually requires additional post filtering such as averaging several consecutive measurements or applying a non-linear filter, such as the median filter to cope with the spurious zero crossings. Another solution is to apply validation equations that would reject a measurement if its value differs dramatically from the previous valid frequency estimate.

Another issue that concerns this approach is the interaction between the frequency estimates and frequency tracking mechanisms. If the frequency tracking mechanism adjusts the sampling frequency, an error may be introduced in a subsequent frequency estimate. The safest solution for the zero-crossing detection algorithm is to use time stamps of the samples instead of the sampling frequency because it may change between the two zero crossings.

4.2 Phasor-Based Methods

The voltage and current phasors estimated from quantized samples by using the relaying algorithms rotate at their radian frequency as shown in Figure 4.2. This feature allows for the measurement of frequency from consecutive estimates of the phase angles. The generic equation for this method is as follows.

$$f(t) = \frac{1}{2\pi}\frac{d\varphi}{dt} \tag{4.2}$$

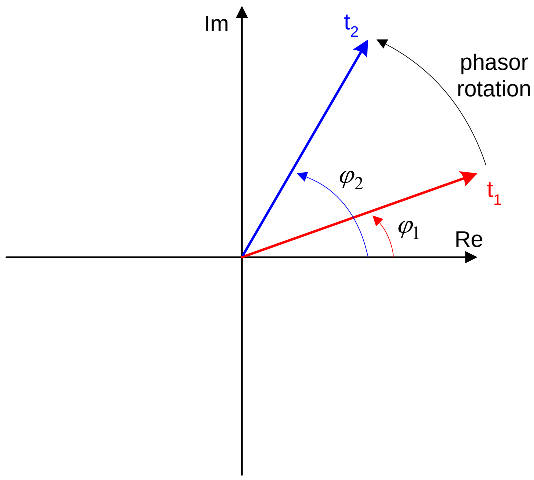


Figure 4.2: Phasor rotation for measuring frequency

This method is applied in different forms; the variations includes the use of differences of phase angles over different time spans, use of different phasor estimation techniques, use of different signals, such as phase voltage or positive sequence voltage, and use of different techniques for filtering the calculated estimates.

This method can update the frequency estimate with each new calculation of the phasor of a voltage or current. Theoretically, it can be done several times a cycle by using the voltages and currents and their combinations. The faster the method of estimation, the higher is the susceptibility to estimation errors.

The interaction between the changes in sampling frequency and the frequency estimates may create some problems. The Fourier transform, and several other techniques assume a constant sampling rate. In case the sampling frequency changes from its nominal value, the number of samples in the data window would not span an integer number of periods of the signal. This would result in errors in the phasor estimates that would result in errors in the frequency estimates. Strategies to deal with this problem include a method that adjusts the length of the data window according to the actual frequency so that the window always covers exactly one period. This approach keeps the length of the data window fixed but dynamically adjusts the digital filters to follow the actual frequency, changes the sampling rate so that the data-window is always of integer number of periods and uses a software re-sampling approach that re-calculates the values of the samples before estimating the phasors.

4.3 Other Methods

Several other algorithms for estimating power system frequency have been presented in the literature. They include approaches such as a linear regression with adaptive on-line re-adjustment of the input filter, least error squares technique, a Newton-type algorithm, and a Kalman filter based approach.

4.4 Frequency Tracking (Small Deviations)

Algorithms for estimating phasors are tuned to a pre-selected nominal frequency (50Hz or 60Hz). At the selected frequency, the gain of the phasor estimator is one. This means that the magnitude of the input signal is measured accurately if the frequency of the waveform from which the samples are taken is the same as the selected frequency. In case the frequency of the waveform is different from the pre-selected frequency that was used to design the phasor estimator, the calculated value of the phasor will oscillate between the lower and upper envelopes as shown in Figure 4.3.

The goal of frequency tracking is to modify the phasor estimation process in such a manner that the phasor estimates remain correct even if the system frequency deviates from its nominal value. This is done by measuring the frequency and adjusting the sampling clock, the algorithm or the values of the quantized samples.

A simple correction for the frequency error is not possible because the error changes with time. One method, however, provides for a unique frequency response in which the upper and lower envelopes are the same for any frequency.

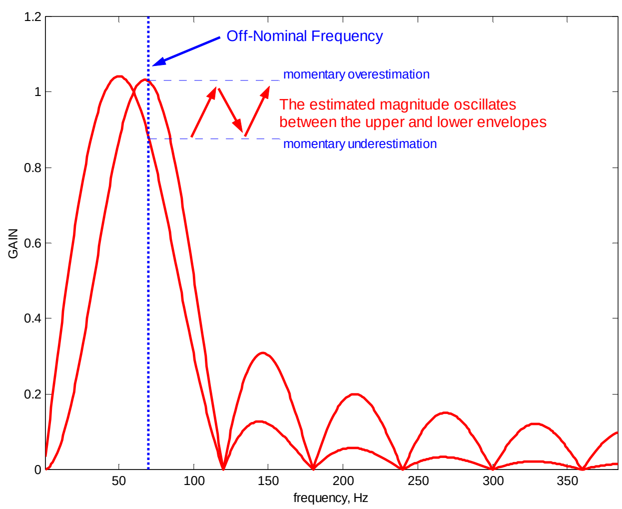


Figure 4.3: Effect of off-nominal frequency on phasor estimation

Conditions, which usually affect the frequency tracking, are noise, spurious zero-crossings, fast frequency changes, sub-synchronous oscillations and power swings.

4.5 Frequency Tracking (Large deviations)

In many applications, off-nominal frequency is to be measured over a wide range. An example of this necessity is the frequency of a thermal unit during the warm phase when the speed of the unit increases from 50% to 100% over a period of several hours. Two specific approaches have been proposed and used for the purpose of measuring frequency over the 30-60 Hz range (25-50 Hz in Europe and some other places).

The first approach is to design an algorithm that would measure frequency with acceptable accuracy over this range. One such example is the algorithm proposed in Reference 15. The second approach is to have software that would implement an approach similar to that described in the flow chart given in Figure 4.4.

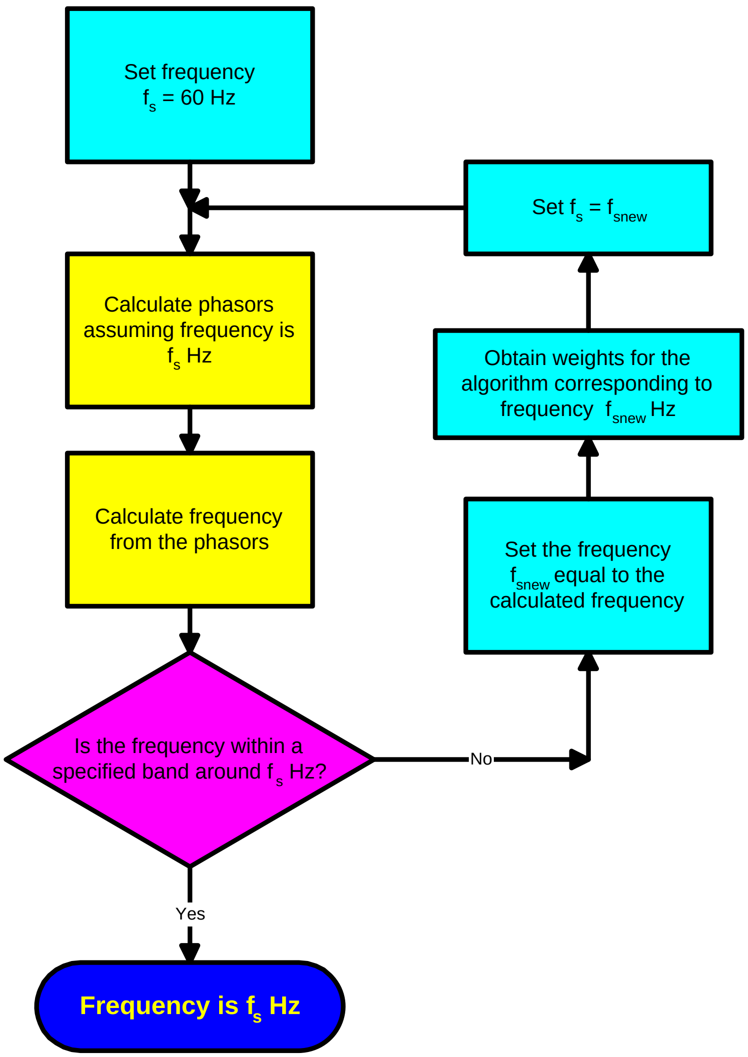


Figure 4.4: An iterative process for calculating an off-nominal frequency

4.6 Phase Angle Displacement between Signals

While calculating the phase angle displacement between two signals appears to be a straightforward problem, it might require some thought. Two methods are described in this section. The first method calculates the phase difference from the instantaneous values of the signals and the second method calculates the phase difference from the calculated phasors.

4.6.1 Instantaneous values method

If the instantaneous values of the samples and the time (at which they are acquired) are available, the zero crossings of the waveforms can be calculated. The time difference between the zero crossings of two waveforms can be converted to phase angle if the waveforms are of a single frequency. This is not usually true and, therefore, it is preferable to calculate the phase angles from the phasors calculated by one of the techniques previously described in this report.

4.6.2 Phasor method

Three scenarios are examined in this category. Consider the first scenario in which different waveforms are sampled and quantized by a relay, and the quantized values are then used to calculate the phasors representing those waveforms. In this scenario, it is assumed that the same algorithm is used to calculate the phasors. The phase angle displacement is the difference between the calculated phase angles representing the waveforms.

Consider the second scenario in which the phasors are calculated by different relays that are installed in the same substation and the master clock of the substation triggers the sampling of the waveforms. In this case also, the phase angle displacement between signals is the difference between the angles of the calculated phasors.

Now consider the third scenario in which the relays are installed in one substation, have clocks synchronized with the GPS or a similar time signal and each relay controls its waveform-sampling process. In this case the sampling in the relays is not synchronized and, therefore there is a skew between the calculated phasors. Each phasor calculated by the relays is time tagged. The phase displacement between the waveforms is the difference between the angles of the calculated phasors corrected for the angle due to the difference in the time tags of the phasors. This is demonstrated in Figure 4.5.

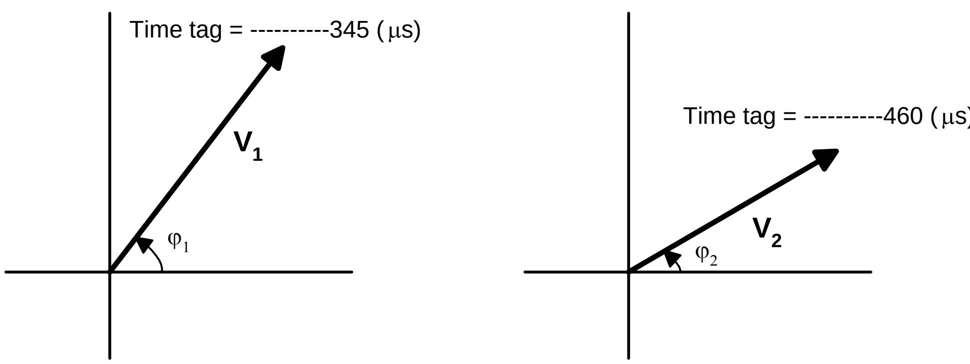


Figure 4.5: Two voltage phasors computed by two different devices

Now consider that the calculated voltage phasors and time tags are as follows.

$$\begin{aligned} &V_1 = 1.05\angle 50.126^o \qquad V_2 = 0.97\angle 31.043^o \\ &t_{v1} = ...345\mu s \qquad t_{v2} = ...460\mu s \end{aligned} \tag{4.3}$$

The phase difference between the calculate phase angles of the phasors is 19.083°. The time tags indicate that the phasor of voltage $\mathbf{V_2}$ was calculated 115 μs after the phasor of voltage $\mathbf{V_1}$; this means that $\mathbf{V_2}$ lags by an additional angle of $115 \times 10^{-6} \times 360 \times f$ degrees. If the frequency is 60 Hz, this comes out to 2.484°. The phase displacement therefore is 21.567°. This is the phase difference between the phasors and not the phase displacement between the zero crossings of the waveforms.

5 TIME DOMAIN ALGORITHMS

The algorithms based on electrical properties of system elements are classified as Time Domain or Modeling Algorithms. Several algorithms have been proposed over the years. In most cases, these algorithms determine faults on transmission lines and transformers. A few samples of such algorithms are described in this section.

When a fault occurs on a transmission line, the voltage on the line depresses and, therefore, the line charging currents are minimal. If the line charging is neglected, the line can be modeled as a resistance and inductance connected in series. In the transformer model, the magnetizing currents and the hysteresis-loss currents are negligible compared to fault currents. The transformer can, therefore, be modeled by the resistances and inductances of the transformer windings and the flux linkages with the windings. The voltage-current relationships of the equivalent circuits expressed as differential equations can be used for detecting faults in transformers. These algorithms are described in this section.

5.1 Line Protection Algorithms

The series R-L circuit of Figure 5.1 can be used to model a line that is experiencing a fault. This model neglects the line charging currents. There are two advantages of this model. The first advantage is that it measures the inductance of the line from the relay location to the fault; this parameter is independent of the system frequency and, therefore, the measurements are valid at the nominal as well as the off-nominal frequencies. The second advantage is that the dc offset in the current does not affect the measurements.

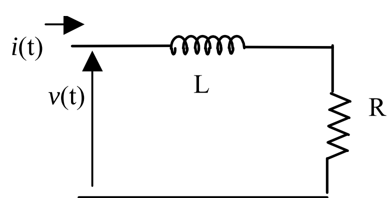


Figure 5.1: Model of a line experiencing a fault

The voltage at the relay location can be expressed as a function of the current in the circuit as follows

$$v(t) = Ri(t) + L\frac{di(t)}{dt} \tag{5.1}$$

Integrating both sides of this equation from time t_0 to t_1 and t_1 to t_2 provides the following two equations.

$$\int_{t_0}^{t_1} v(t)dt = R\int_{t_0}^{t_1} i(t)dt + L\big(i(t_1) - i(t_0)\big) \tag{5.2}$$

$$\int_{t_1}^{t_2} v(t)dt = R\int_{t_1}^{t_2} i(t)dt + L\big(i(t_2) - i(t_1)\big) \tag{5.3}$$

Using the trapezoidal rule for integration and assuming that the time interval between consecutive samples is small, the integrals from sample *k* to sample *k+1* and from sample *k+1* to sample *k+2* can be evaluated. Rearranging the integrands provides the following two equations in two unknowns, which are the resistance and inductance from the relay location to the fault.

$$R = \left[\frac{(v_{k+1} + v_k)(i_{k+2} - i_{k+1}) - (v_{k+2} + v_{k+1})(i_{k+1} - i_k)}{2\left(i_k i_{k+2} - i_{k+1}^2\right)} \right] \tag{5.4}$$

$$L = \frac{\Delta T}{2} \left[\frac{(v_{k+2} + v_{k+1})(i_{k+1} + i_k) - (v_{k+1} + v_k)(i_{k+2} + i_{k+1})}{2\left(i_k i_{k+2} - i_{k+1}^2\right)} \right] \tag{5.5}$$

Notice that the process uses three sets of samples (the data window is three samples in this case). The denominator, $2\left(i_k i_{k+2} - i_{k+1}^2\right)$, of Equations 5.4 and 5.5 is not constant but varies in time with local minima at points where both the current and the derivative of the current are small. This is shown in Figure 5.2.

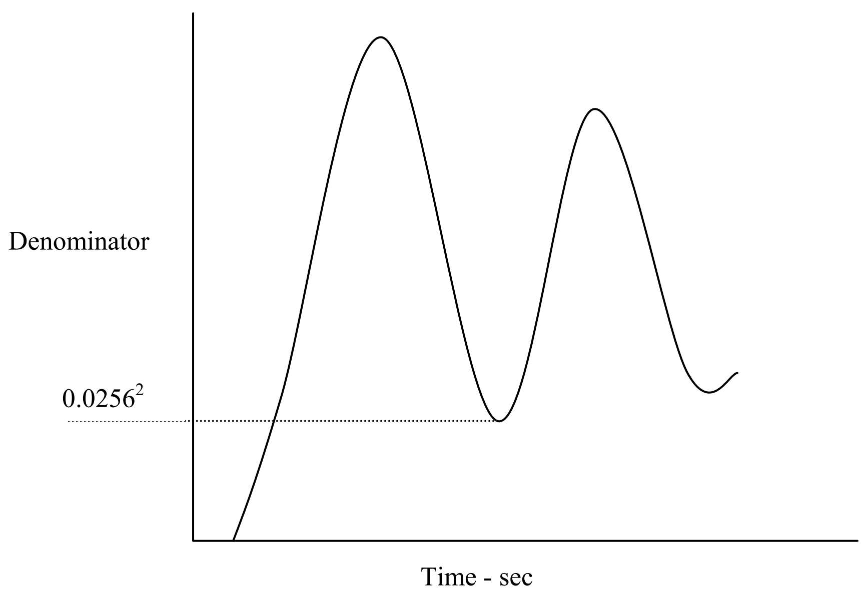


Figure 5.2: The denominator of Equations 5.4 and 5.5

The data window using this algorithm can be extended by using one of the following techniques.

1. Counting scheme
2. Use of median filter
3. Improved integration of currents and voltages

5.1.1 Counting scheme

A counting scheme can be used to ensure that the trip decisions are secure. A possible approach is to set a counter to zero and increment it by one if the calculated R and L lie in the operating zone of the relay and decrement it by two if the calculated R and L lie outside the protection zone of the relay. Consider that a trip decision is implemented if the counter exceeds six. If the data is sampled at 960 Hz (16 samples per cycle), the earliest trip decision would take one-half cycle. Each estimate that lies outside the protection zone would delay the decision by two additional samples.

5.1.2 Using median filter

A median filter is implemented by ranking the inputs by their amplitudes and selecting the middle value as the output. A median filter that has five inputs, x[n-2] to x[n+2], the output y[n] would be

$$y[n] = \text{median}\{x[n-2], x[n-1], x[n], x[n+1], x[n+2]\} \tag{5.6}$$

These filters isolate noise spikes while they preserve the essential features of the inputs.

5.1.3 Improved integration of currents and voltages

The differential equation that describes the R-L model of a transmission line is given in Equation 5.1. The integration of this equation, expressed by Equation 5.2, can be rewritten as follows.

$$\int_{t_0}^{t_1} v(t)dt = R\int_{t_0}^{t_1} i(t)dt + L\int_{t_0}^{t_1} \frac{di}{dt}\Big|_{t=t_0} dt \tag{5.7}$$

The derivative of a sinusoid is a sinusoid that has a magnitude increased by a factor ω and advanced by 90°. If three samples are used to implement the integration by using the trapezoidal rule, the process will have a frequency response given by the following equation.

$$D(\omega) = \frac{j\omega}{f_s}\left[0.5e^{-\frac{j\omega}{f_s}} + 1.0 + 0.5e^{\frac{j\omega}{f_s}}\right] \tag{5.8}$$

However, the discrete-time implementation of the integration of a derivative is usually of the form i_{t1}-i_{t0}. The frequency response of this process, D_{eq} is given by

$$D_{eq}(\omega) = e^{\frac{j\omega}{f_s}} - e^{-\frac{j\omega}{f_s}} \tag{5.9}$$

The frequency responses of the processes defined by Equations 5.8 and 5.9 differ as shown in Figure 5.3. The response of the process represented by Equation 5.9 matches the desired response only for a small band in the low frequency region. Signals, which contain components of frequencies in the range where the desired and achieved responses differ, will not be accurately computed by the process described by Equation 5.9 causing errors in the estimates of R and L. It is, however, possible to design an FIR filter that would implement the integration of the derivative and would have the desired frequency response. Figure 5.4 shows the frequency response of a filter that uses seven samples. This filter has the desired frequency response in the

$0 \le \frac{f}{f_s} \le 0.4$ range. The improved integration technique can be used when two or more than three samples are used to implement the integration of the derivative.

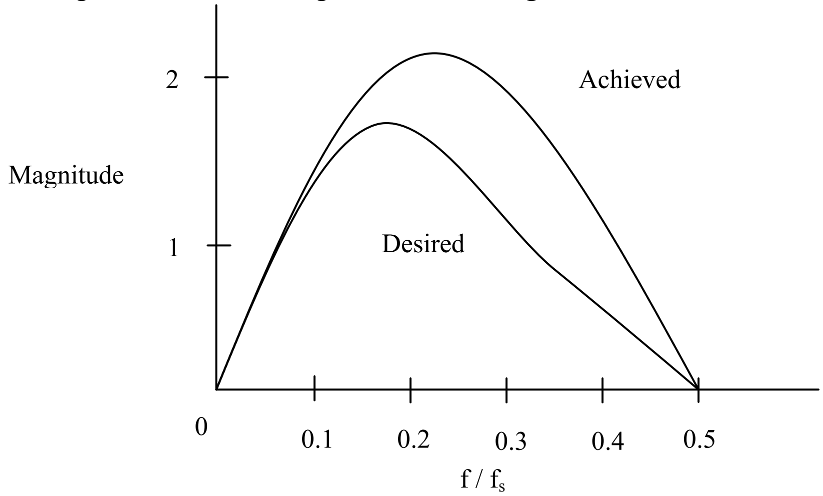


Figure 5.3: The desired and achieved frequency responses of the integration of the derivative by using a three-sample FIR Filter

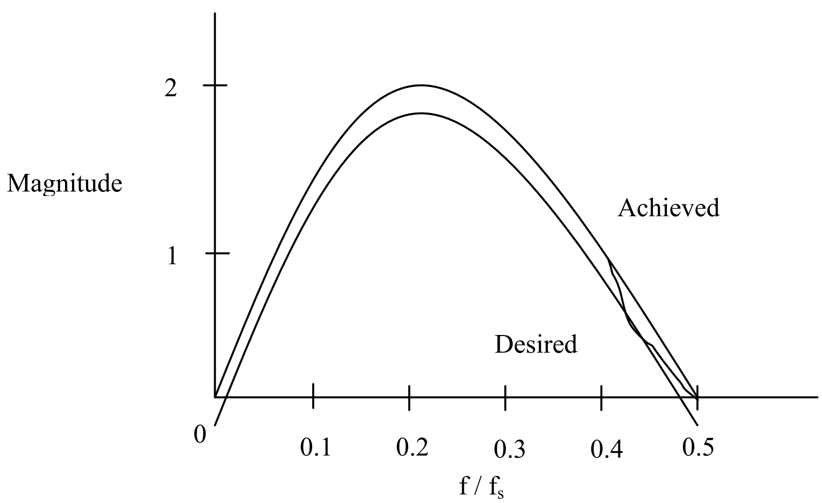


Figure 5.4: The desired frequency response and achieved frequency responses of the integration of the derivative term by using a seven-point FIR Filter

5.2 Differential Equation Algorithms for Transformer Protection

These algorithms model a transformer in its normal operating condition and match the operating state of the transformer with the model to distinguish faults in the transformer from external faults. One of the approaches is described in this section with reference to a single phase transformer and is then applied to a three phase delta-wye transformer. Consider a two winding single phase transformer shown in Figure 5.5.

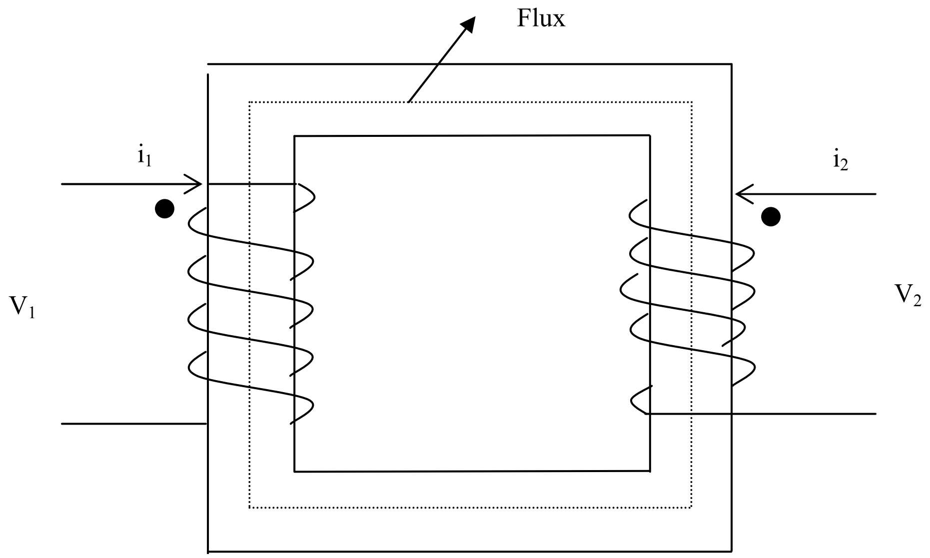


Figure 5.5: Electromagnetic circuit of a single - phase transformer

The primary voltage, v_1 may be expressed as a function of the current i_1 in the primary winding, the resistance r_1 and leakage inductance l_1 of the primary winding and the mutual flux linkages Λ as follows.

$$v_1 = r_1 i_1 + l_1 \frac{di_1}{dt} + \frac{d\Lambda}{dt} \tag{5.10}$$

Similarly, the voltage v_2 can be expressed as a function of the current, i_2, in the secondary winding and the resistance r_2 and the leakage inductance l_2 of the secondary winding and the mutual flux linkages as follows.

$$v_2 = r_2 i_2 + l_2 \frac{di_2}{dt} + \frac{d\Lambda}{dt} \tag{5.11}$$

Equations 5.10 and 5.11 can be combined to eliminate the mutual flux linkages; this provides the following equation

$$v_1 = v_2 + \left(r_1 i_1 - r_2 i_2\right) + \left(l_1 \frac{di_1}{dt} - l_2 \frac{di_2}{dt}\right) \tag{5.12}$$

Integrating both sides of this equation from t_1 to t_2 provides

$$\int_{t=t_1}^{t=t_2} v_1 dt = l_1\left[i_{1(t_1)} - i_{1(t_2)}\right] - l_2\left[i_{2(t_1)} - i_{2(t_2)}\right] + \left[\int_{t=t_1}^{t=t_2} v_2 dt\right] + r_1\left[\int_{t=t_1}^{t=t_2} i_1 dt\right] - r_2\left[\int_{t=t_1}^{t=t_2} i_2 dt\right] \tag{5.13}$$

If the time from t_1 to t_2 is one sampling interval, ΔT, and the samples taken at time t_1 are identified by the subscript k, the samples taken at time t_2 are identified by the subscript k+1 and the trapezoidal rule is used to integrate the terms in Equation 5.13, the following equation is obtained

$$\frac{\Delta T}{2}\left[v_{1(k)}+v_{1(k+1)}\right]=\frac{\Delta T}{2}\left[v_{2(k)}+v_{2(k+1)}\right]+l_1\left[i_{1(k+1)}-i_{1(k)}\right]-l_2\left[i_{2(k+1)}-i_{2(k)}\right] \\ +\frac{\Delta T}{2}r_1\left[i_{1(k)}+i_{1(k+1)}\right]-\frac{\Delta T}{2}r_2\left[i_{2(k)}+i_{2(k+1)}\right] \quad (5.14)$$

This equation expresses the integration of the primary voltage as a function of the secondary voltage, currents in the primary and secondary windings and the transformer parameters. Because the transformer parameters are known, and the primary and secondary voltages and currents are sampled and quantized, the left and right-hand sides of Equation 5.14 can be calculated at every sampling instant. During normal operation, external faults and magnetizing inrush, the calculated values of the left and right-hand sides of Equation 5.14 would be equal for all practical purposes but will not be equal during faults in the protection zone of the transformer.

5.2.1 Application to three phase transformers

The technique described in the previous section can be extended to protect banks of three single phase transformers and wye–wye connected transformers. However, when delta-wye transformers, shown in Figure 5.6, are to be protected, the technique has to be adjusted because the currents in the delta-winding are not usually available.

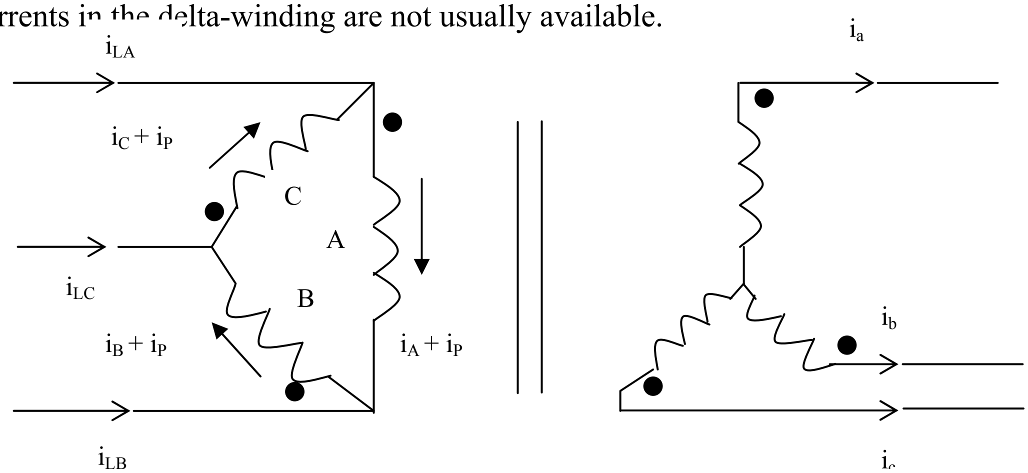


Figure 5.6: Electrical connections of a three phase delta- wye transformer

In this case, the currents in the delta winding have two components, the circulating currents and the non-circulating currents. When the currents in the delta winding of the transformer can be measured, the technique described in the previous section can be applied. If it is possible to measure the currents in the lines leading into the delta winding only, the non-circulating components of the currents in delta windings can be obtained.

The procedure described earlier to the phase A-a of the transformer can now be applied. This leads to three equations; the zero sequence current circulating in the delta winding is one of the unknowns in these equations. One of the three equations can be used to estimate the circulating current and the other two equations can be used to verify if the transformer is experiencing a fault or not. Details of this procedure are given in Appendix B.

5.3 Flux Restraint Algorithm for Transformer Protection

The terminal voltage can be expressed by the following equation if the voltage drop in the resistance of the transformer winding is neglected.

$$v(t) - L\frac{di(t)}{dt} = \frac{d\Lambda}{dt} \tag{5.15}$$

Applying the trapezoidal rule of integration to this equation leads to

$$\int_{t_1}^{t_2} v(t)dt - L\left[i(t_2) - i(t_1)\right] = \Lambda(t_2) - \Lambda(t_1) \tag{5.16}$$

Integrating the voltage using the trapezoidal rule and rearranging this equation provides

$$\Lambda(t_2) - \Lambda(t_1) = \frac{\Delta T}{2}\left[v(t_2) + v(t_1)\right] - L\left[i(t_2) - i(t_1)\right] \tag{5.17}$$

Now consider that the time from t_1 to t_2 is one sampling interval, ΔT. Also, express the samples taken at t_1 by the subscript k and express the samples taken at t_2 by the subscript k+1. Equation 5.17 now becomes

$$\Lambda_{k+1} = \Lambda_k + \frac{\Delta T}{2}\left[v_{k+1} + v_k\right] - L\left[i_{k+1} - i_k\right] \tag{5.18}$$

If the initial flux Λ_0 were known, then Equation 5.18 could be used to update the flux from current and voltage measurements. The flux–current characteristic would be as shown in Figure 5.7. If the initial flux is not known, then the flux–current characteristic has the correct shape but is liable to be shifted up or down as shown in the figure. The flux-current characteristic during an internal fault is shown as a dashed line.

The difficulty of not knowing the initial flux can be resolved by considering the following equation.

$$\left(\frac{d\Lambda}{di}\right)_k = \frac{\Delta T}{2}\left[\frac{v_k + v_{k-1}}{i_k - i_{k-1}}\right] - L \tag{5.19}$$

The slope of the flux linkages during the operation in the unsaturated region of the magnetization curve, shown in Figure 5.7, is large whereas the slope during internal faults and during operation in the saturated region is small. The algorithm differentiates between internal faults (the slope is consistently small) from magnetizing inrush (the slope alternates between large and small values). A counting scheme can be used to stabilize the protection application. The counter increases if the slope is less than a threshold and the counter decreases if the slope is greater than the threshold.

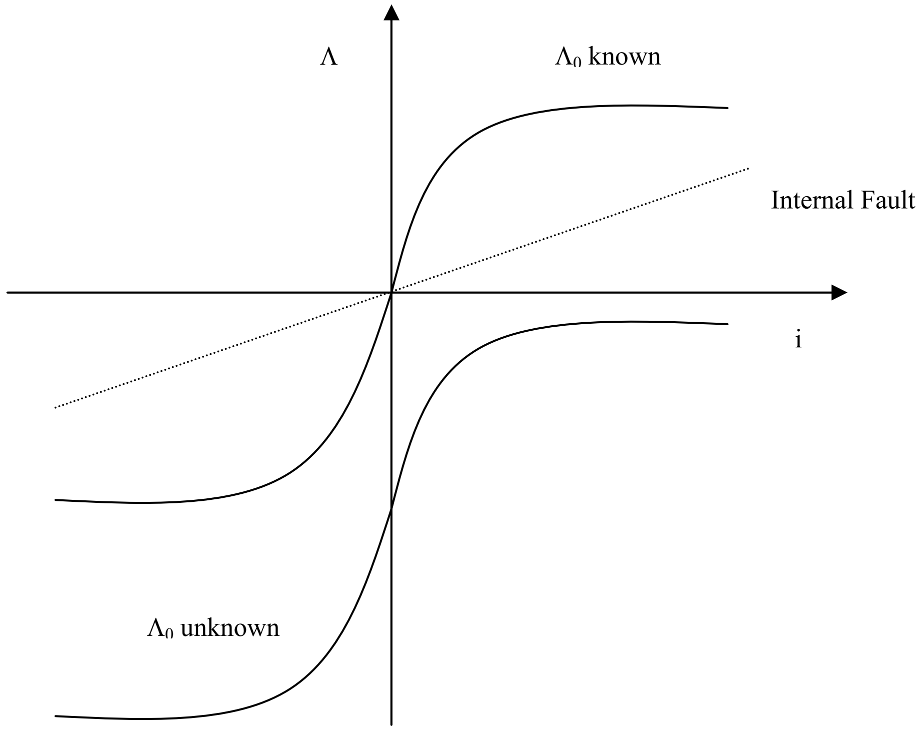


Figure 5.7: The flux-current characteristics when the initial flux is or is not known

6 DATABASE ISSUES

6.1 Data Recording

Most numerical relays being marketed at this time have built-in waveform and event recording capabilities. The relays energize the trip circuits causing the circuit breaker(s) to open for isolating the faulted element. The quantized values of samples of voltages and currents are usually saved in circular buffers that are sufficiently large to record data from the pre-fault, fault and post-fault periods.

It is desirable to obtain waveform information by using sampling rates higher than the sampling rates used by the protection-algorithms. The voltages and currents are, therefore, sampled at a higher rate and the quantized data is decimated to lower rates for calculating phasors or for implementing other procedures used for generating trip decisions. The operating speed of a relay depends more on the size of the sampling window than the sampling rate. The sampling window size is usually one cycle and, therefore, the average 'information delay' is one-half cycle.

The process of decimation is more complex than simply 'taking every Nth sample' if the decimation factor is N. If a proper decimation filter is not used; aliasing would occur when the data is processed by the relay.

Some considerations regarding the data recording function are:

- *Frequency response* — The highest harmonic that recordings can "see" depends partly on the sampling rate, and partly on the effects of filters used in the data collection path.
- *Record length* — Total record length and pre-fault record length are important
- *Triggering* — A local trip initiates a recording, but recording could also be initiated manually or through logic input to the relay.
- *Record storage* — Memory for storing recordings is limited, beyond which new data is overwritten on the old records
- *Settings files* — A settings-file attached to each recording helps in analyzing 'what happened?' since the relay settings might well have changed since the recording was made
- *Software* — The software used to view the recordings should be user-friendly.
- *Off-nominal frequency* — The sampling rate may automatically 'track' the actual frequency in order to keep the number of samples per cycle constant. This can be a problem when a recording is compared with other recordings whose sampling rates are different.
- *Time stamp* — A satellite system, such as IRIG-B, may be used to help where comparisons between recordings is desired

6.2 Managing Relay-Setting Files

Before the use of microprocessors for preparing and maintaining relay-setting databases, they were maintained in simple forms. The databases usually consisted of relay types, purpose for which they were used, the locations where they were installed and their settings. The information was usually stored at the central engineering office where the data were maintained by those responsible for their development. The relay-databases were important because they provided quick overview of the settings being used and made it easy to change relays and implement changes in their settings. The databases were able to generate printed instructions for changing relay settings that field personnel could follow. Maintaining a single database by an electric power utility made it easy to search for specific relay types and other common information.

With the advent of numerical relays, the number of settings of a relay varies from a few to several hundred. Many settings are determined by the software provided by the manufacturers. This has made it almost impossible to maintain the databases as was being done earlier while the need for maintaining databases increased substantially. Some manufactures provide software that includes a database for managing the relay settings. There are two problems with this approach. Firstly, the software is revised frequently and, sometimes, the revised versions are not backward compatible. Moreover, some manufacturers provide different software for different relays. Secondly, software can handle the settings of the relays of that manufacturer only. This would be fine if a utility used relays of the same manufacturer only. These problems forced the utilities to develop new approaches for preparing and maintaining relay-setting databases.

There are several approaches that can be used for developing databases. One possible approach is described in this section as an example. This database (like many other databases do) can be used for setting up a file management system for numerical relays. This approach uses a file structure for storing the files of settings on a central computer. The file structure and databases used by utilities work as the file management system.

6.2.1 Developing a file management system

6.2.1.1 Linking with the existing databases

It is desirable to continue using the existing relay-settings databases that include information on electromechanical, solid-state as well as numerical relays. The existing databases can be linked to an overall database system that would include information on all relays.

6.2.1.2 Including additional information on numerical relays in the traditional databases

The additional information needed to successfully setup a file management system that includes numerical relays is described in the following paragraphs.

Equipment: The equipment such as transformer, breaker, line, circuit, or bus that the relay is protecting should be recorded if it is not already being done. This field would contain information such as, TX#1, Bus#1 or Line 23.

Relay identification by its function: The relay identification by its function, in the form of identification numbers assigned in the IEEE Std. C37.2-1996, IEEE Standard Electrical Power System Device Function Numbers and Contact Designations should be included for facilitating the searching and accessing relay data. The identification numbers could be extended to include additional information. For example, a differential relay protecting Transformer 1, may be identified as 87T1; 87 identifies it to be a differential relay and T1 identifies it as a relay protecting Transformer 1.

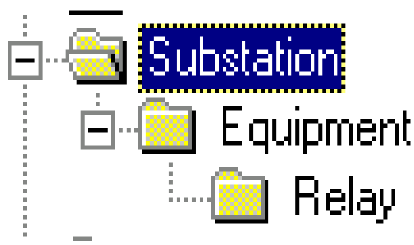
Software for loading the settings: The software and its version used to create the settings-file should be recorded. This would assist the users to select the appropriate software for updating the files containing information on the settings. This is a very important field and will be even more valuable in the future.

Settings-file name: The name of the settings-file, which would be used to link with the database, should be recorded. Another possible approach would be to store the directory path. Some database software, which can be used for storing relay information, allows the database records to be linked files.

This is the minimum information that should be stored. Additional information, such as the actual settings, could be included but these are already recorded in files with the help of the settings software and are used for managing the relay settings. Maintaining duplicate records would be a lot of work with very little advantage if any. However, it is advisable to store a backup copy of the database in a different computer.

6.2.2 File directory structure

A simple file structure for storing the settings-file for each relay separately is desirable. The directory structure and naming convention are discussed in this section:



6.2.2.1 Substation

The name of this directory should be the name of the substation (or the abbreviation of the name) where the relay is located and should match the name recorded in the relay database.

6.2.2.2 Equipment

The name of this directory should match the name of the equipment that the relay is used to protect. For example this directory might be named as TX#1, Bus#1 or Line 23.

6.2.2.3 Relay

Each relay protecting a component should have its own directory. This would allow prompt access to the records without the need for memorizing the directory names where the information is stored. The name of the directory should match the Relay Functional Identification indicated in the database for that relay. This would tie the relay directory to the database as well as to the functional diagram of the relay. A user could look at a functional diagram and go to the directory that contains the settings-file. An example of a directory name might be 87T1 for the transformer differential relay provided to protect Transformer 1.

6.2.3 Settings-file name

In most utilities, a technician working in a substation would download the settings-file from a remote location. Naming the settings-file for identification purposes should be done with care. A possible approach is to use the substation name concatenated with the functional identification of the relay. For example, a file name for the settings of a transformer differential relay provided in the substation PSRC would be PSRC_87T1.ext. In this approach, the underscore separates the substation name from the functional identification. This would help the user to easily identify the file downloaded from the protection engineer's office. Depending on the limitations of the software, the length of the file name should include as much of the substation name as possible. The extension of the file name would depend on the manufacturer of the relay.

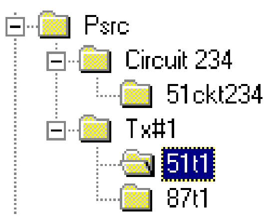
6.2.3.1 Example

Consider a substation called PSRC that contains the following relays and equipment.

1. Transformer 1 is protected by a differential relay that has the functional identification of 87T1. The transformer backup protection is provided by an overcurrent relay that has the functional identification of 51T1.
2. Circuit 234 is protected by an overcurrent relay that has the functional identification of 51ckt234

The directory structure and file names for the relay-settings could be as shown here.

Directory Structure **Filename**



PSRC_51ckt234.ext

PSRC_51t1
PSRC_87t1

6.2.4 Security issues

The users of the files should be allowed access only for reading the files. However, access to read/write should be given to those responsible for the development of settings-files.

6.2.4.1 Testing-files

If automated testing is used, the test-files can also be added to the relay directory using the same naming convention. The extension will usually distinguish them from the settings-files. If both the settings and testing-files are stored, they could be zipped into one file that would reduce the time needed to download them from a remote location.

6.2.4.2 'Back From Field' files

It is necessary to ensure that the settings-files are up-to-date. A procedure must be established for returning the files with final settings to the database management system when any settings are changed. The file created in the field and returned to the database management system should be reviewed and then posted as new settings-file for future use.

6.2.4.3 Issuing changes

When relay settings are to be changed, a copy of the settings-file should be made and changes should be posted in the file. The name of this file could be the same as the original file with a P added at the beginning of the name or the word pending at the end of the name. This file would be downloaded by the technician for changing the relay settings.

6.2.4.4 Conclusion

The approach described in this section is in use in a utility and works very well. The description provided in this section shows that the files can be organized for ease of access. Moreover, the described approach takes advantage of the databases and the relay functional diagrams already in use by the utility.

6.3 Configuration Tools

Modern microprocessor-based relays provide the user with significant flexibility. Configuration mechanisms available on modern microprocessor-based multi-function relays include the features described in this section.

6.3.1 Numerical settings

This is a basic configuration that is available in all protective relays. The practice, in the past, was to make available to the user numerous parameters of a protection element. The practice has continued in the numerical relay technology.

6.3.2 User programmable curves

Numerous protection functions, such as time overcurrent, under- and over-voltage and volts-per-hertz, use inverse time characteristics. Typically, several standard curves are provided, and a user can select the characteristic needed for an application. In many numerical relay designs, a user is permitted to specify a few characteristics on a point-by-point basis for better coordination of the protection function.

6.3.3 Multiple operating modes

Often several modes of operation of a numerical protection are available to the user. For example, an out-of-step tripping element may generate a trip command instantaneously in an “Early mode” or generate the trip command slowly in a “Delayed mode” when the current levels drop so that the interrupted current is less than the rupturing capacity of the circuit breakers. Another example is that the neutral directional overcurrent element may be polarized from the zero sequence voltage, ground current, or both. A time overcurrent element may respond to the true RMS value (appropriate for thermal protection) or to phasor magnitude (appropriate for overcurrent protection), etc. This level of flexibility in the form of a multi-choice selection was not available in the previous technologies.

6.3.4 Configurable analog inputs

Some numerical relays can be set to respond to different input signals. This is a brand new level of flexibility that is now available at the time of configuring the relay. The mechanism when available on a relay is referred to as “Analog Points” or “Sources”. For example, a transmission line relay for a breaker-and-a-half configuration, which measures the two currents, allows the basic protection functions, such as distance protection to respond to the sum of the two currents and also to provide for two breaker fail functions.

6.3.5 Programmable logic

Programmable logic has become a standard feature for microprocessor-based relays. It permits a user to perform basic control functions and to build an application from elements available in a relay. The outputs from the protection elements are combined with auxiliary signals for use within the relay and for interfacing with the output contacts or for sending them to other local or remote equipment over communications channels.

A typical set of functions consists of gates, latches, timers, edge detectors, and counters. Some relays allow a user to attach a "digital element" to internal signals within the programmable logic scheme for naming the signal and generating display messages or sequence of events logs.

Advancements in relay communications, such as the IEC 61850 protocol and GOOSE messaging, allow the exchange of binary information on a peer-to-peer basis between several relays. This has opened a whole new area for distributed logic schemes.

6.3.6 Configurable output contacts

Configurable output contacts provide the convenience of driving the contacts from any variable within the programmable logic. This has become an industry standard as well.

6.3.7 Configurable display messages, target LEDs and event logs

Numerical relays allow the user to specify for each protection and control element whether a target LED should be fired and/or an event log should be created upon pickup, dropout or operation of the element. This allows the customizing of the relay and avoids overflowing SOE records by non-critical information. User-programmable LEDs and display messages have also become an option. This capability is used to customize and enhance the relay from the local display point of view.

6.3.8 Programmable oscillography

Numerical relays allow the user to generate oscillography records. The user's choices could include the selection of sampling rate, file content, triggering signal, position of a trigger (split between the pre-trigger and post-trigger data), number of records versus record length, ways to treat old records (overwrite or not), etc.

6.3.9 Multiple setting groups

Multiple (switchable) setting "groups" (or "banks") feature has become an industry standard. A modern numerical relay allows the users to enter several values of the same setting, typically organizing the entries into groups, and provide a programmable mechanism to switch the groups based on specified conditions, such as state of protection and control elements, input contacts, keypad commands, communications ports, self-monitoring alarms, etc. This feature, in association with programmable analog inputs, can be used to

(a) apply a single relay to different system configurations, such as a line with normal and bypass terminal switching arrangements and

(b) perform switching of CT circuits in software instead of hardware.

The use of multiple setting groups to perform "adaptive" protection functions is likely to increase in the near future.

6.3.10 Firmware and software version number

Relay firmware and software are often revised to remove deficiencies, add new features, improve performance and resolve issues requested by users. The need for firmware and software version control is becoming more apparent as questions such as the following arise:

- Is one device with a particular version of firmware compatible with another (same) device with a different version of firmware?
- Are devices with older firmware versions compatible with a newer software interface?
- Are the existing settings used on the device still applicable if the firmware is upgraded?
- How will performance be impacted if the firmware is not changed?
- How can the firmware and software changes found?
- How can the severity of a firmware and software change be evaluated to determine if it is necessary to change or not change the firmware?
- How can the documentation be kept up-to-date as changes are made?

These, and many more questions like them, illustrate some of the challenges in dealing with changes to the firmware or relay interface software. IEEE Standard C37.231 “Recommended Practice for Microprocessor-Based Protection Equipment Firmware Control” can help both users and manufacturers to deal with this issue. This recommended practice suggests the appointment of a specific person from the user organization (the firmware controller) that could be responsible for maintaining records and correspondence with the manufacturers regarding any changes to the device firmware or associated software. The recommended practice also specifies that manufacturers should associate a level of importance with any change and indicate what the severity of the change is and the implication of changing the firmware (or not). The importance of accurate record keeping with respect to device model and serial numbers and the associated firmware version and software version is also highlighted in C37.231.

7 SPECIAL PROCESSES

7.1 Time Division Processing

A multifunction microprocessor-based relay can have many protection, control, metering, and communications functions that seem to operate simultaneously. Generally, a single processor is provided to execute the code and it can only perform one task at a time. This affects the performance of a numerical multifunction protective relay. The designers take this into consideration while structuring the embedded firmware so that the power system protection functions are executed in a timely manner.

7.1.1 Processing resources

The processing resources available in a microprocessor-based relay are finite. The resources depend on the speed and architecture of the processor (or processors) and the speed and architecture of the peripherals, such as memory, data bus etc. Relay designers select (hopefully) the best compromise in allocating the finite processing resource to create a multifunction relay. Performance and functionality of a relay can be increased by upgrading the processing power; but this generally comes at a lot of expense.

Some of the important factors that a relay designer tries to balance in the design are as follows:

- *Number of functions included*: If the relay has fewer functions, the code for each can be processed more often resulting in less delay in generating an output for each function.
- *Frequency of processing functions:* The more often a task is processed, the more processing resources are used by that task; this reduces latency between the occurrence of an event on the power system and the instant when the relay makes a decision.
- *Efficiency of code*: If the code is written in a high level language, such as C, it is easier and faster to write and debug; but, the code is not as efficient as a well designed code in assembly language.

7.1.2 Structure of embedded firmware code

The performance of a relay depends on how busy it is because it performs tasks in a serial processing manner. For example, if a function is not in a picked up state when its sub-routine is called, the processing may jump to the next task that is to be executed. However, if the function is in a picked up state, the relay may have additional tasks to do such as determine how close to timed out the function is or set target bits, etc. before going to the next task. The result is that the time to execute each routine could depend on the state of the power system. Also, when a power system disturbance occurs and multiple elements are picked up, it takes longer to process each task causing the relay to take longer to make a trip decision. When the embedded firmware is structured to process all tasks serially, the performance can be non-deterministic.

There are ways that relay designers use to structure the code to deal with this situation. Some of the approaches used are discussed in the following sections.

7.1.3 Fixed interrupts

Typically, the firmware program includes fixed interrupts for doing some tasks, such as acquiring data from the power system. These tasks must be performed at specific intervals regardless of what other tasks the relay might be performing. When a fixed interrupt occurs, the microprocessor stops performing the routine that it was performing, does the time critical task and then resumes the non-critical task where it left off.

7.1.4 Triggered interrupts

One way that a relay designer can provide high performance for certain critical tasks is to monitor an indicator condition. For example, the currents and voltages could be monitored for a sudden change by a disturbance detector function. If a disturbance is detected, then an interrupt is issued to stop the relay from performing non-critical or background tasks and start processing the critical protective functions immediately.

7.1.5 Fixed Interval processing loops

Another way that the relay manages the processing resources is to set up several processing loops. For example, the code may be structured such that an A/D interrupt runs at the pre-specified sampling rate. The processing of the sampled data for calculating the operating quantities, for example phasor estimates, may not be done every time a new set of samples is acquired. These high-burden tasks would be done as often as the highest priority tasks are called. The high priority instantaneous trip functions and output logic functions might be executed in a $^1/_8$ or ¼ cycle loop. Another group of lower priority functions might be processed only every cycle or every other cycle. Non-protection functions may only be processed every ¼ to 1 second intervals. Finally, there may be a background processing loop that is only processed when extra time is available after all fixed priority tasks have been completed. This scheme ensures that the relay performance can be targeted as needed and can be determined.

When different functions are processed at different rates, it is important that the dependencies between functions be carefully analyzed. A function that is updated every ¼ cycle may be missed by a function that is only updated once per cycle. Or, if one function relies on the output of another function, there is no point in processing the dependent function more often than the independent function.

7.1.6 Processing overload

When the code is structured into many processing loops that are assigned different priorities or are initiated with different interrupts, some tasks may not be completely processed if the code cannot be executed completely before the next interrupt comes around. If this occurs, it is difficult to predict what the relay would do. It is, therefore, necessary to include a multitasking operating system that monitors all of the tasks and interrupts and provides a relay trouble alarm if all of the tasks are not being completed in the allotted time. The operating system may also mitigate the situation by putting off non-critical tasks during system disturbances until the power system returns to a normal steady state operation.

7.1.7 Post disturbance processing

The relay designer can ensure that the processor is not overloaded during power system disturbances because of storing information concerning non-critical tasks during a disturbance instead of rescheduling them for execution after the power system returns to a normal operating state. For example, calculation of distance to fault can be a high processing burden task. It is not critical that the results be available until some time after the fault has been identified and isolated. The targeting of such functions for post fault processing leads to better performance of the relay.

7.2 Logic Implementation

Multifunction microprocessor-based relays are typically designed to provide primary protection for a specific type of power system equipment, for example a transformer or a transmission line. The relays include a set of specific protection elements, such as a differential protection and a thermal overload protection of a transformer or a multi-zone distance protection of a transmission line. They also include a set of additional protection elements, such as directional or non-directional phase, ground or sequence overcurrent elements, under and over voltage elements, breaker failure protection and other protection or non-protection functions.

The outputs of different protection functions and the status of relay inputs can be combined in fixed logic schemes that meet the requirements of typical applications for the specified type of protection. For example, receiving of a permissive signal from the remote end of a line combined with the pickup of the Zone-2 distance element, in a permissive overreaching scheme, would generate a trip for the line circuit breaker. Typically, such schemes are developed and tested by the relay manufacturers and the users have to enable them and have to set the required parameters that determine the implementation of the scheme.

The use of fixed schemes is very effective; however they cannot meet the requirements of all special cases and the philosophies implemented by different utilities. This is where programmable scheme logic comes into play.

Most relays include programmable scheme logic (PSL - PL is more popular) with different levels of complexity. In some low-end relays, it is limited to masks that allow the assignment of relay inputs and outputs to a specific protection function. In advanced relays, the programmable scheme logic includes multiple logic gates of different types, multiple timers, ability to invert signals and many other features.

The PSL can be used for

- mapping of opto-isolated inputs, relay output contacts and programmable LED's.
- enabling the development of customer specific scheme logic.
- providing facilities for conditioning relay outputs (delay on pick-up/drop-off, dwell time, latching or self-reset).
- defining the protection elements or conditions that would trigger the start of a Fault Recorder.

The main purpose of the programmable scheme logic (PSL) is to allow the relay user to customize an individual protection scheme for meeting the requirements of specific applications. This is achieved through the use of programmable logic gates and delay timers.

The inputs to the PSL are a combination of the status of digital signals from the opto-inputs, outputs of the protection elements and outputs of the fixed protection scheme logic. The outputs of the PSL are the relay output contacts, programmable (if any) LEDs and virtual outputs that define a data set transmitted over the substation local area network.

The PSL can be executed on predetermined time intervals or can be event driven. The event driven mode is more efficient since the logic is processed whenever an input provided to it changes. To further improve the efficiency, only the part of the PSL that is affected by the status change is processed.

The PSL provides significant improvement in adapting a microprocessor-based relay to different applications. Because PSL is developed by the user from the tools provided by the manufacturer, it requires adequate testing to prove that it will perform as intended when a fault or an abnormal system condition occurs.

7.3 Self Monitoring

Microprocessor-based relays have the capability of performing Self-Monitoring or Supervision. The extent of monitoring depends on the type and manufacturer of the relay. The response of the relay to a Self-monitoring alarm depends on the type of error. This section provides as overview of the typical self-monitoring provided in microprocessor-based relays.

Self-monitoring is typically performed separately on different components of a relay. Figure 7.1 illustrates the typical architecture of a microprocessor-based relay.

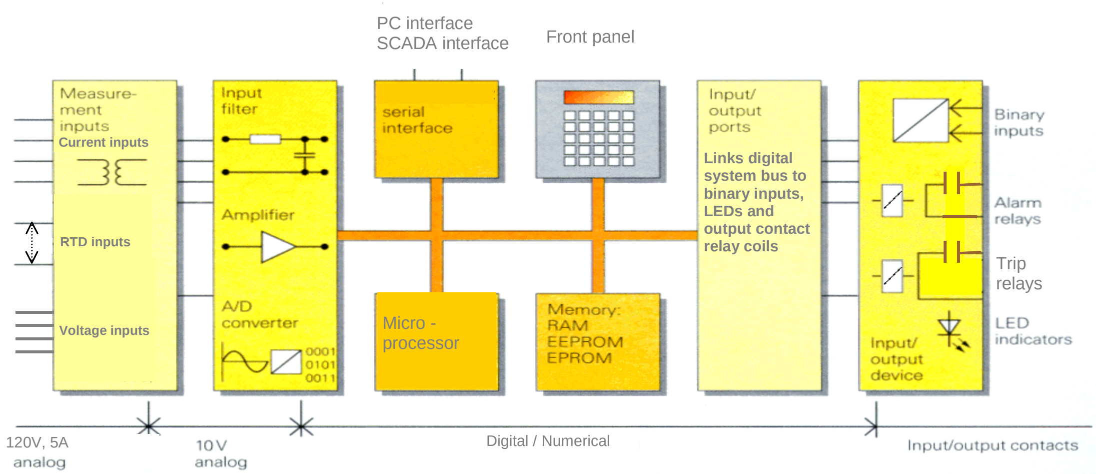


Figure 7.1: Components of a typical microprocessor-based relay

Working in sequence from inputs through to outputs, the different categories of supervision performed by relays could consist of the following functions.

A. Input circuit measurement

A.1. Current monitor

A.1.1. **Current symmetry** is used to check for valid current measurement. If a symmetry threshold is exceeded then an alarm is generated. Such an alarm indicates a problem in the current measurement. When this alarm occurs without a current plausibility error (see A.1.2), the anomaly is from a broken CT connection or compromised CT.

A.1.2. **Current plausibility** (summation) also indicates a current measurement problem and is alarmed. If observed without a current symmetry error (see A.1.1), the anomaly is in the A/D converter.

A.2. **Voltage monitor**

A.2.1. **Voltage unbalance** is used to check for valid voltage measurement. An alarm indicates a problem in the voltage measurement. When this alarm occurs without a voltage plausibility error (see A.2.2), the anomaly is from a broken PT connection or compromised PT.

A.2.2. **Voltage plausibility** check can be done when a broken delta connection is connected to the relay in addition to the 3 phase voltages. If observed without a voltage unbalance error (see A.2.1), the anomaly is in the A/D converter.

A.2.3. **Phase rotation** checks are performed. An alarm here is useful especially during commissioning to ensure that the connections are correct.

A.2.4. **Voltage level** measurement is used to check all 3 phases are correctly connected. It can also be used to block related protection functions (for example, distance protection - 21, ac directional relay - 67) to prevent false tripping.

A.2.5. **Fuse Failure Monitoring** is used on PT circuits protected by fuses. This monitoring is achieved using voltage and current measurements. Assertion of this function may be used to block voltage dependent elements (for example, distance protection - 21, ac directional relay - 67).

A.3. **Binary Inputs** can be monitored by plausibility checks with related measurements. An example of a binary input supervision is the monitoring of 52a and 52b contacts together with current measurements. A discrepancy showing an open circuit breaker with current flowing is alarmed.

B. **Station battery/supply voltage monitoring** is performed by measuring the supply voltage and issuing an alarm when the value is above or below the set-point.

C. **Relay battery** alarm measures the voltage of the relay battery. This battery is used to back-up data stored in RAM for example, waveform capture records. An alarm here does not normally block any protection functions.

D. **DC/DC converter supervision** is achieved by monitoring the electronic component supply voltages (typically 5V, 15V and 24V DC). An alarm ensures that the relay does not issue a trip command should the voltage vary beyond the tolerances of the electronic components.

E. **Data acquisition (A/D Converter) monitoring** is carried out by applying a known reference voltage and comparing it to the measured numerical value. Should there be a difference between these two values, the relay measurement circuits are blocked to prevent a false trip.

F. **Software/Program** monitoring checks the operation of the microprocessor and the software.

F.1. **Watchdog functions** detect if the software performs given tasks within the allocated time. This is achieved by starting a timer at the beginning of a routine and is only reset at the completion of the routine. An alarm here blocks the relay.

G. **Memory checks** are performed using check-sum calculations. The different types of memory are handled separately. An alarm here blocks the relay.

H. **Outputs** have limited or no monitoring. The output ports and control circuits may be monitored. The trip, close and alarm contacts are not monitored. The health of the individual contacts can be checked through functional testing.

I. **Communication interfaces** may have monitoring depending on the type of communication protocol used. These may be used in distributed bus protection configuration to ensure that all remote bay units are communicating with the master unit.

J. **Displays, keypads and LEDs** are typically not monitored.

A live status contact (normally closed contact) is generally available on digital relays. When the relay is healthy this contact is held open. A monitoring alarm (including a loss of power supply) releases the relay coil and a normally closed contact closes. This facility is used to indicate a potential relay problem.

7.3.1 Summary

Self-monitoring in modern protective relays helps in improving system reliability (through improved availability) and reduces maintenance costs by reducing the need for routine testing. Self-monitoring, however, does not cover all connections and components, but gives digital relays a major advantage over the solid-state and electromechanical relays.

Relays do not monitor circuit breakers, although trip coils and continuity of trip coil circuits can be monitored. There may still be the need to periodically operate circuit breakers to ensure that they mechanically operate as expected.

8 OUTPUTS

8.1 Tripping Local Circuit Breakers

The operation of the protective relay is not complete without the clearing of the fault condition. The interface of the relay with the switching devices or any other equipment in the substation is typically achieved through the relay outputs.

The number of relay outputs is a very important criterion in the selection of a specific microprocessor based relay, since it will determine the overall functionality of the device and even the performance of the protection system.

For example, if the relay is a transmission line protection device that is connected to a breaker-and-a-half substation and the utility is using single-pole trip and reclosing, only these functions will require a total of 12 relay outputs. If the relay initiates two trip coils and breaker failure is a built-in function, the number of the relay outputs will significantly increase.

If the number of relay outputs is limited, auxiliary relays might be used to meet the requirements of the application. However, this will introduce an additional time delay that in some case might not be acceptable for stability or power quality reasons.

8.2 Tripping Remote End Circuit Breakers

Tripping of remote end circuit breakers requires the availability of a communications channel between the two (or more) substations.

The tripping of remote end circuit breakers can be executed in several different ways, depending on the capabilities of the protection device and the available communications interface with the remote end.

The most typical case is by operating a relay output of the relay that is wired to an opto-input of a communications device. It will detect the energization of the input and will convert this status change into encoded data packet that is sent to the remote end of the line. There the message is decoded and results in the operation of a relay output of the communications device. This output is wired to an input of the receiving IED at the remote substation. When the input is energized, based on local supervision or not, it will result in the tripping of the local breaker.

As can be seen from the description of this method, the signal goes through many transformations that will result in a time delay that might not be acceptable.

That is why many relays with communications capabilities, such as line differential relays, allow the sending of direct or permissive inter-tripping signals to the remote end of the line as part of the encoded message transmitted to the remote end with analog data. This leads to a significant improvement in the operating time of the scheme in the cases when remote end breaker tripping is required.

8.3 Messages to Regional and System Control Centers

The multifunctional microprocessor based relays are becoming the main source for analog and binary status information in integrated substation protection and control systems. They monitor the status of their associated breakers based on 52a or 52b contacts or both (in the cases when the reliability of the status information is important to the application).

The change of state information is used for many different purposes such as adaptive protection or to initiate certain operating procedures at the regional or system control center.

The development of GOOSE (Generic Object Oriented Substation Event) messages in IEC 61850 standard allow the sending of high-speed peer-to-peer communications messages to the substation computer or HMI. That information is then sent to any client interested in it.

The GOOSE message is based upon the asynchronous reporting of an IEDs digital outputs status to other peer devices enrolled to receive it during the configuration stages of the substation integration process. Since it is used to replace the hard wired control signal exchange between IEDs for interlocking and protection purposes, it is mission sensitive, time critical and must be highly reliable.

The associated IEDs receiving the message use the contained information to determine what the appropriate protection response is for the given state. The decision of the appropriate action to GOOSE messages and the action to take should a message time out due to a communication failure is determined by local intelligence in the IED receiving the GOOSE message. It can be used for Breaker Failure Protection to initiate a distributed Breaker Failure Protection function or to trip the adjacent breakers.

To achieve a high level of reliability, messages will be repeated as long as the state persists. To maximize dependability and security, a message will have a time to live, which will be known as "hold time". After the hold time expires the message (status) will expire unless the same status message is repeated or a new message is received prior to the expiration of the hold time.

The repeat time for the initial GOOSE message will be short and subsequent messages have an increase in repeat and hold times until a maximum is reached. The GOOSE message contains information that will allow the receiving IED or substation client to know that a message has been missed, a status has changed and the time since the last status change.

In order to achieve high speed performance and at the same time reduce the network traffic during severe fault conditions, the GOOSE message has been designed based on the idea to have a single message that conveys all required protection scheme information regarding an individual protection IED. It represents a state machine that reports the status of the devices in the IED to its peers. To allow further customization of the GOOSE messages, individual applications can map other status points to the User Defined bit pairs UserSt (details of User bit Defined pairs are given in IEC 61850).

8.4 Data for Relay Engineers

Besides relaying, microprocessor-based relays also provide metering, sequential event reporting, and monitoring functions. The relays also provide information on their settings upon request.

Microprocessor-based relays collect and analyze data to determine whether or not to operate. Most of the data along with certain decisions made by the relay are available to protection engineers. The data includes the time of the disturbance, currents and voltages in phasor and (or) oscillographic forms, the zone in which the fault is estimated to be, the location of the fault, the phases involved, the relay logic response and operating time, and reclosing and relaying-channel performance.

Fault location is only as good as the information provided to the relay. If the total line distance is inaccurate, then the fault location will be just as inaccurate. Many users of microprocessor relays opt to enter the line distance as 100%, thus getting fault reports in percentage of the line. Faults on tapped lines cannot be distinguished from faults on the main line and the apparent impedance seen and reported by the relay may differ from the actual impedance to the fault. Wire or construction changes on the line may also introduce errors in the fault location. In spite of all these limitations, information on fault location provided by the relays usually proves valuable.

In addition to retrieving information from the relays provided on the faulted line, relay engineers can retrieve information from other, backup relays to ensure that they had also seen the fault, but were either properly restrained or their operation was adequately delayed. The availability of all this information makes it possible to measure the reliability of microprocessor-based relays.

Microprocessor-based relays provide metering functions. The currents and voltages can be retrieved from the relays at any time, not just during disturbances. They can report on instantaneous and peak demands. They can provide information on all phases and positive, negative and zero sequence quantities.

Because microprocessor-based relays know the fault current, the relays can contain a function to monitor the cumulative current interrupted over time. This information can be further converted to the level of breaker contact wear. By looking at the difference between when a trip signal was sent and the time that the current went to zero, microprocessor relays can provide breaker timing functions.

Microprocessor relays perform routine self-checks and report any problems. They can also monitor and report on external conditions that may affect their operation, such as temperature, power supply and channel availability. If the problems are deemed severe, the relays can remove themselves from service.

9 TESTING

Microprocessor relays can also be tested with the apparatus used for testing analog and electromechanical relays. Figure 9.1 shows a typical set up for testing a relay. The signal source could be a relay test set, playback simulator or a real time simulator.

In comparison to the older relays, microprocessor relays usually have graphical user interfaces (GUI's) to help in setting the relays and in giving information on the trip events. This information would include identification of elements that had tripped and when they had tripped as well as provide records of the waveforms that caused the trip.

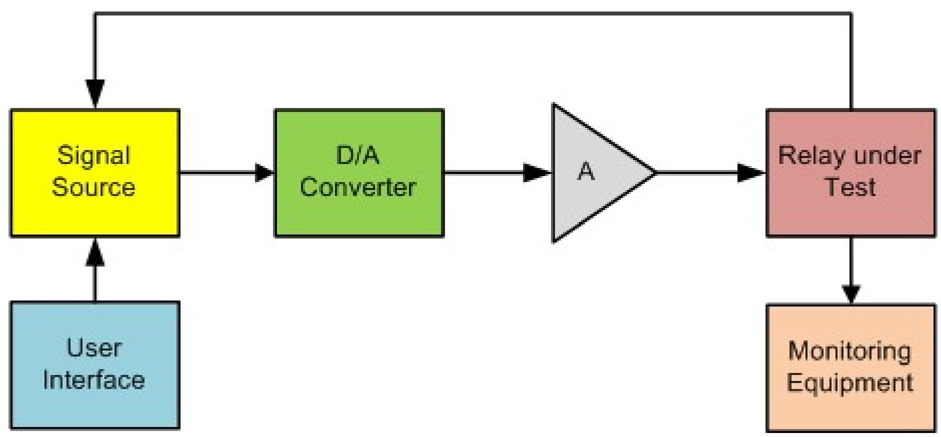


Figure 9.1: Diagrammatic layout of test apparatus

Some of the latest microprocessor relays have a facility which allows the test signal to be applied at the A/D level in the relay (±10V). Bypassing the auxiliary CTs and PTs provided in the relay has the advantage of not requiring the conditioning amplifiers shown as "A" in Figure 9.1. The output signals from the D/A converters in the signal source can be taken into the A/D stage of the relay. The test apparatus can be much smaller and lighter if the conditioning amplifiers are not required. On the other hand, this kind of testing will not pick up any problems in the auxiliary CTs and PTs due to saturation or over-voltages.

It is simpler to simulate microprocessor relays than it was to simulate the older analog relays. If a simulation model of the relay is available, the application scenarios can be investigated by incorporating the model in an EMTP simulation. This kind of testing is interactive because the model output (trip signal) can be used to open the requisite breakers in the simulated power system and the downstream effects (sympathy trips) can be examined. This is not a substitute for testing an actual relay but can be used to investigate application issues involving multiple relays.

In summary, the test techniques are the same as for older relays but microprocessor relays have some additional features, which give more information about relay behavior in test situations.

9.1 Non-real-time Testing

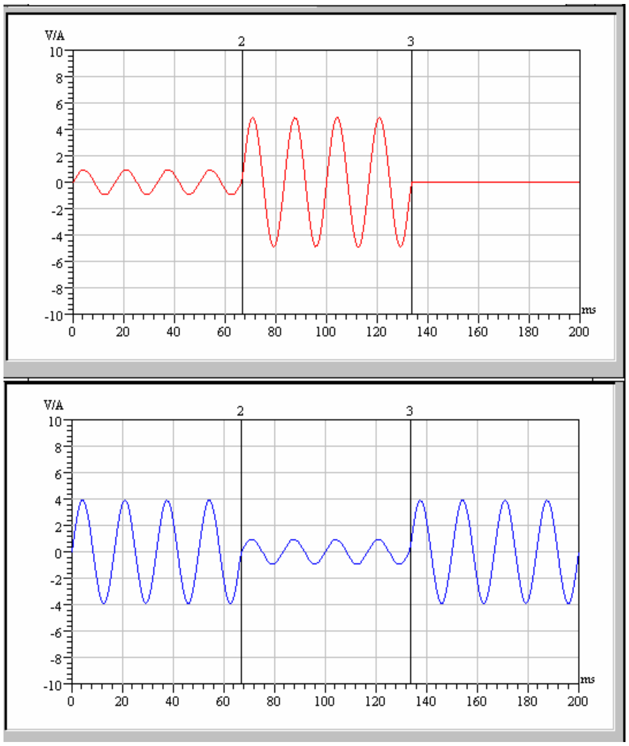


Figure 9.2: Three state changes on a current and voltage channel

Figure 9.1 shows the type of circuit used to connect a relay to the test apparatus. The signal source can be one of the following types.

- *A relay test set*
- *Playback digital simulator*
- *A real-time digital simulator*

In most cases, the D/A converter is already built into the signal source and the user interface is via a PC plugged into a serial port. The conditioning amplifiers, A, are usually separate from the signal source. The action of the relay may be recorded, or used, by the test set as well as the monitoring apparatus. The relay test set and the playback simulator are non-real time test devices in that the waveforms used to test the relay are not being calculated in real time by simulations of the power system to which the relay is connected but have been calculated and stored prior to the test. The typical modern relay test set has up to 12 analog low level (±10 volts) channels suitable for driving conditioning amplifiers to produce the correct levels of signals for applying to the relay. A test set usually has a number of opto-isolated digital outputs to simulate signals, such as incoming pilot signals from a remote source. The configuration of the outputs is set up through a GUI running on a laptop plugged into, or forming part of, the test set. The GUI, which probably runs under Windows, allows the relay engineer to access the various test functions and to set the signals on each channel for the desired test. The signals can be viewed on a monitor screen prior to applying them to the relay. They can be set to begin outputs at a precise time using a GPS timing reference.

Test sets are capable of performing what are known as

1. *Steady state tests*
2. *Dynamic tests*
3. *Transient tests (in playback mode)*

The control features of a test set can allow each channel to be set independently for simulating unbalanced conditions. Additional dialog boxes allow the test engineer to add harmonics and/or exponential dc components to specific signals. The waveforms can be sent to the output channels for providing inputs to the conditioning amplifiers when the engineer is satisfied with the waveforms on each channel (as previewed on the screen).

For *dynamic state tests*, it is necessary to build two or more "states" of suitable duration to represent the chain of events. Generally, the events start during normal load conditions, change abruptly to the sine waves representing the steady state signals representing a fault and then change abruptly to the post-fault sine waves. The transient, that represents the change from one state to the other, is not included in the signals. This type of testing may provide uncertain or incorrect results if used with high speed relays. Figure 9.2 shows this situation for two channels with the upper channel representing the currents and the lower channel representing the voltages.

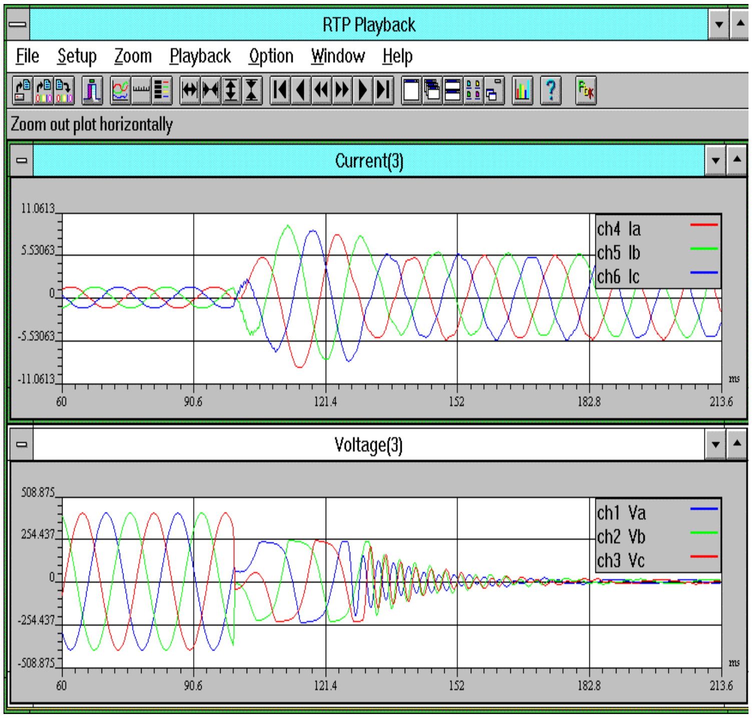


Figure 9.3: Typical playback screen

When the relay test set is used in the playback mode, it uses pre-recorded data files that represent the system events. The data files can also come from recordings taken by digital fault recorders, relays, or emtp simulations. The format of the files may be in accordance with the IEEE Std. C37.111, Standard Common Format for Transient Data Exchange for Power Systems (COMTRADE) but could be specific to the test set manufacturer. The playback software should allow the files to be displayed and edited for dealing with missing data or obviously wrong data. When the test engineer is satisfied with the displayed data, the waveforms can be applied to the relay. A typical waveform display is as shown in Figure 9.3. It contains a few cycles of pre-fault waveform followed by the fault waveform. (The waveforms in the display are from a series compensation study with the MOV action clipping the voltage waveforms followed by ring-down when the bypass closes.) Depending on the length of the recording the waveform may go to a post fault value but in most cases, particularly where the file is generated by an emtp, it will consist of only two states, namely, the pre-fault and the fault. The playback simulator should preferably be capable of looping around the first pre-fault period in order to allow the engineer to set up all the apparatus in the pre-fault steady state and should only play out the recording on command when the engineer is satisfied that all the recording apparatus etc. is ready and any memory features in the relay are well established. At the end of the file, the simulator should then loop around the last fault cycle for a specified time to allow such things as zone 2 timers to time out. It is important to limit the number of loops around the last cycle to prevent damage to the apparatus under test. This would be a feature chosen by the engineer before initiating the playback.

Playback simulators are mostly used for examining the first relay action that occurs during a disturbance. This would normally be the primary protection of the relay nearest the fault. It is possible to use successive playback tests to investigate downstream actions following the operation of the breaker tripped by the primary protection but this can be a tedious process. For this type of investigation it is much better to use a real time simulator which can accept the trip signal from the relay, open the appropriate circuit breaker(s), and continue the simulation with the network in the new state. It is becoming increasingly important to examine downstream events following a correct primary operation. “Sympathy trips” can lead to widespread cascading trips particularly on parallel lines where the opening of a breaker on the faulted line causes phase reversals in the un-faulted line.

9.2 Real-time Testing

When a relay is being tested as part of a system, with the effect of relay action being an essential feature of the investigation, interactive testing using a real time simulator is desirable. Until the 1990’s, the only real time simulators were analog model power systems (MPS) and analog Transient Network Analyzers (TNA). Real time digital simulators are now available. These simulators are much smaller and cheaper than their older analog counterparts.

When a real time simulator is available, there is no need to store waveforms other than for display purposes. The instantaneous values calculated at the end of each time step are provided. (Present technology uses time steps of the order of 50µs.) The relays under test can interact with the simulation by opening a breaker on the simulator and the simulation continues with the breaker open. A recloser may then reclose the breaker, and so on, until the system settles down to the post fault steady state. Figure 9.4 shows signals recorded during a phase fault, circuit breaker trip and after recloser. The relay under test tripped and then reclosed the simulated breaker in the real time simulator. The waveforms were recorded on an oscilloscope and show a short fault duration. The circuit breaker pole opened and then closed 100 ms later. The simulation includes a multi-mass turbine generator set which swings against the ideal source at the opposite end of the line. The system being run on the simulator is shown in Figure 9.5. Any external control apparatus, which accepts secondary transducer level signals, can be connected to the simulator via conditioning amplifiers. This includes energy monitors on series compensation capacitors, Power System Stabilizers, relays, reclosers, FACTS controllers, etc.

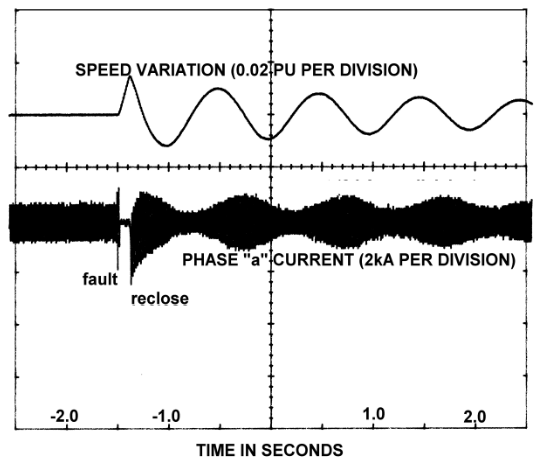


Figure 9.4 Signals from a single pole trip and reclose study

The real time simulator is a much more time effective tool than the playback simulator for studies involving interaction between relays, energy monitors, and controllers etc. but is less likely to be portable although the digital real time simulator is much smaller than the analog simulators and is much simpler to set up.

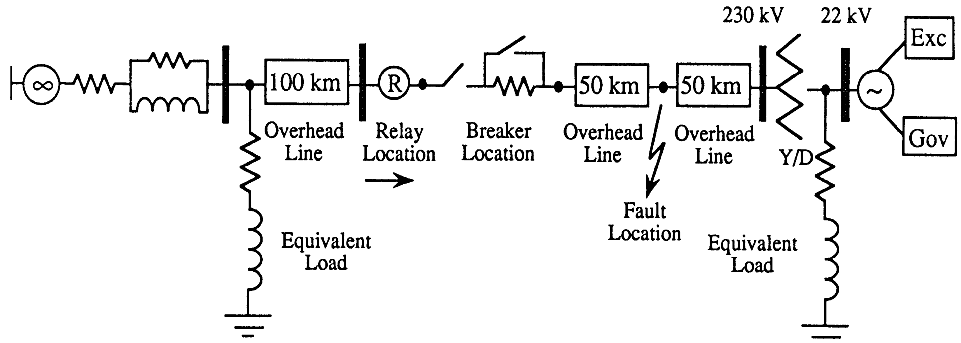


Figure 9.5 One machine/infinite busbar system running on the simulator

9.3 Non-real-time Interactive Testing

This form of testing is only possible if an accurate digital model of the relay is available. It is not a substitute for testing the actual relay but it allows the engineer to examine application issues involving the relay in question. The relay model forms part of the simulated system which is running in an off-line simulation program such as EMTP, EMTDC, ATP. When, on receiving the inputs, the relay model provides a trip signal, it is passed on to the circuit breaker model in the simulation. The circuit breaker then opens and the simulation continues with the network in the new state. If there are relay models at other busbars in the network, they will see the downstream effects of the first breaker opening on the faulted line. This allows the engineer to assess the security of the relay with respect to sympathy trips. If any kind of marginal operation is discovered, tests should be conducted using the actual relay with a real time simulator running the same system. Off-line interactive testing can be used to minimize the time spent on testing on a real time simulator by eliminating all but the marginal cases.

Manufacturers are likely to have accurate transient models of their relays and protection algorithms but do not make them widely available because of the need to protect proprietary information. Generic models of many types of relay are available in the literature and they can be used successfully to explain and investigate application issues. Phasor based models are widely available but they are most appropriately used for slowly varying system events such as power swings, voltage sags, and zone 2 or 3 operations in an impedance relay.

9.4 Testing Future Protection Systems

In a modern substation, relays may be configured to exchange messages with their peers and the substation central computer. These messages may form part of a breaker failure scheme, a System Integrity Protection System (SIPS), a wide area protection feature, etc. To fully test those features relay test methods should include the communication messages in the appropriate format (DNP3, IEC 61850, etc.) as well as the voltages and currents used in a conventional test method. Where the relay is designed to receive digital signals from a merging unit or the IEC 61850 process bus then the test voltages and currents applied to the relay must also be in the correct digital format. Modern test sets and simulators should be capable of producing the test signals in the correct format.

Full scale testing of a SIPS or a wide area protection feature is complicated. Scheme designers need to be aware of any limitations imposed by the need to test the scheme prior to implementing it on the actual system. Even if all the relays/controllers involved in the scheme can be assembled in one location synchronized test signals of considerable time duration have to be injected into each one. If a real time simulator is not available some form of synthetic signal generation is necessary to form a custom test solution for the scheme under consideration.

10 COMMUNICATIONS

Most relays marketed at this time include facilities for them to communicate with its local and remote peers, and with substation and control center computers. The subject of understanding communications for protection engineers is being addressed in a separate report. An overview of communications by Intelligent Electronic Devices (IED), which is the more recently used term for numerical relays now, is provided in this section.

10.1 Communications requirements in multifunctional protection IED's

Each of the functions imposes different communications requirements for the protection IED's. The initial requirements for communications with the early generations of microprocessor based protective relays were driven by the need for setting the relays and extracting fault information and can be divided in two main groups: Local communications and Remote communications

10.2 Local communications for settings and fault records

Local communications are the first type of communications that was required for the earlier generation of microprocessor based relays that started the process of converting the relay into an Intelligent Electronic Device. The main reason for that was the requirement for setting the relays and checking the fault records after a short circuit condition has been detected and the fault cleared.

Some microprocessor based relays had a key pad on the front that allows the entering of settings, however it is a very slow process that is typically used for changes in individual settings. Some relays didn't even have a key pad for settings, so the only way of setting the relay was through a front RS232 serial port.

Another form of local communications is for archiving information. Since older relays had a limited memory that allowed the storage of a single fault record, there was a need to print the fault record, i.e. a parallel printer port was required on the front of the relay as well.

10.3 Remote communications for settings and fault records

Interfacing the relay from the front port however requires the relay engineer or technician to walk to the relay and connect a laptop to the front port in order to change the settings or upload the fault record. This means that it requires physical presence in the substation. However, if the substation is in a remote location and in the middle of a storm, the time that is needed to retrieve the fault record information from the relay will be very long. Someone has to drive to the substation and that might not be acceptable. This is when the requirement for remote communications, the ability to dial-up the relay from a remote location through a modem and ability to change settings or extract fault records, become important.

Fig. 1 shows a basic protective relay with analog current and voltage inputs, opto inputs for monitoring the status of substation equipment or receiving of some forms of control signals, relay outputs to operate breakers or indicate changes of state of the relay, as well as front and back serial communication ports. The above described two serial are still available in modern microprocessor based protection IED's and are also used for extraction of event and disturbance (waveform capture) records.

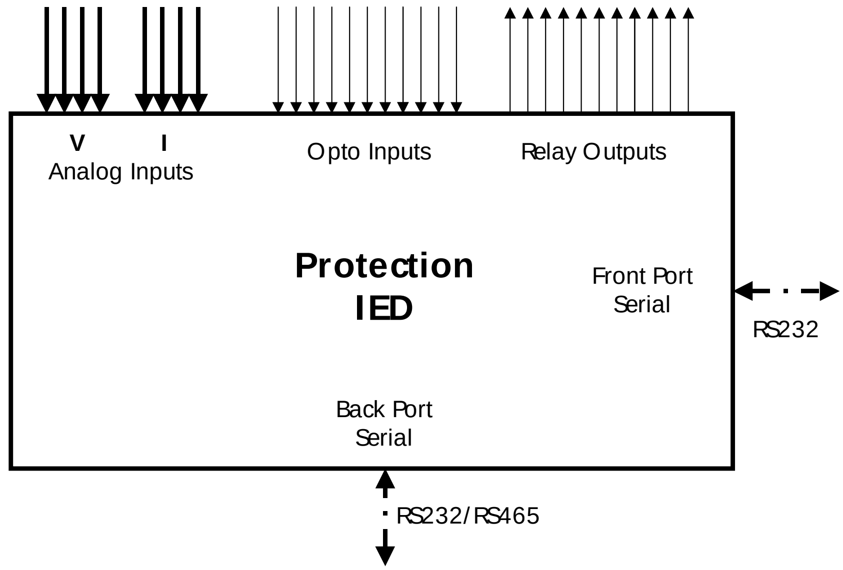


Figure 10.1: Protection IED with simple communications capabilities

10.4 Requirements for communications based protection systems

10.4.1 Line differential protection

Line differential protection has also been one of the early applications of communications in microprocessor based relays. It requires a relatively high-speed exchange of vector information for the currents in three phases (in a segregated line differential relay) between the multiple ends of the line.

A direct fiber optic interface between the relays provides the required high-speed communications, while at the same time eliminates the effects of electromagnetic transients on the data exchanged.

The speed requirements determine the use of full duplex (transmission in both directions simultaneously and requires two pairs of wires), i.e. two communication ports – one for Transmit and one for Receive.

For very critical protection applications the relays may be required to support redundant communications, i.e. a primary and a backup channel that results in the availability of two sets of communication ports (Fig. 10.2).

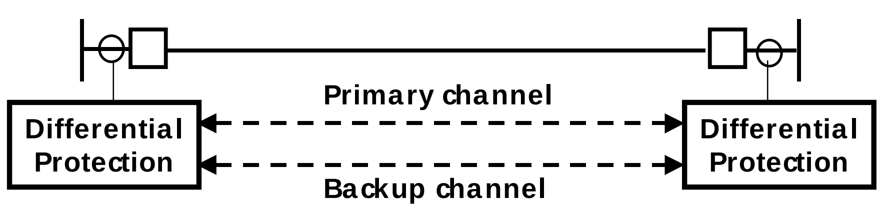


Figure 10.2: Line differential protection with redundant communications channels

Two sets of communication ports are also required in the case of line differential protection application for a three terminal line (Fig. 10.3)

To calculate differential current between line ends it is necessary that the current samples from each end are taken at the same moment in time. With a 64 kbit/s channel typically used to transfer phase current vectors between line ends, it is necessary to measure, and compensate for the propagation delay of the channel.

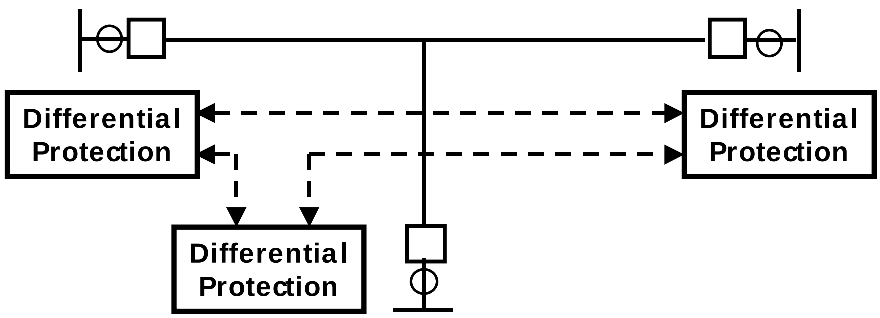


Figure 10.3: Line differential protection for a three terminal line protection application

In the traditional "ping-pong" technique, the relays assume that transmit and receive channels follow the same path and so have the same propagation delay time. This assumption is perfectly valid for relays using a direct fiber connection between line ends. In this case, the length of transmit and receive paths is near identical, messages travel at the speed of light, and thus the propagation delays will be identical (and very small).

Direct fiber communications are often restricted to protecting lines less than 50 miles in length, with a multiplexed telecommunications link more economical for longer distances. Multiplexing allows the available bandwidth of any fiber optic backbone to be better utilized, with relay messages interleaved between telephone, fax and other digital data.

Ping-pong time alignment is still a valid method of compensating for the channel propagation delays, even when multiplexed links are used. The proviso is that transmission time in one direction must be identical with the transmission time in the other direction. Transmit and receive paths for the relay data messages must therefore take the same routing within the telecommunications network.

SDH/SONET hierarchy is becoming more and more commonplace in telecommunications networks worldwide. Such networks can be deployed in flexible, self-healing topologies. Typically, ring network topologies are employed and these are characterized by the ability to self-heal in the event of a failure of an interconnection channel.

In the cases where SDH/SONET systems maintain a split path routing for the Transmit and Receive signals, and as a result, the Transmit and Receive times differ by more than 1ms, the traditional ping-pong technique may not be suitable; this depends on the design of the relay. GPS based time synchronization is the solution for this kind of application that will require the addition of a communications interface between the relay and the GPS receiver in the substation.

Fig. 10.4 shows a line differential relay with multiple communication ports that ensure the dual communication channel requirements from Fig. 10.2 and Fig. 10.3, plus the front and back serial ports for communications based setting changes and fault records retrieval.

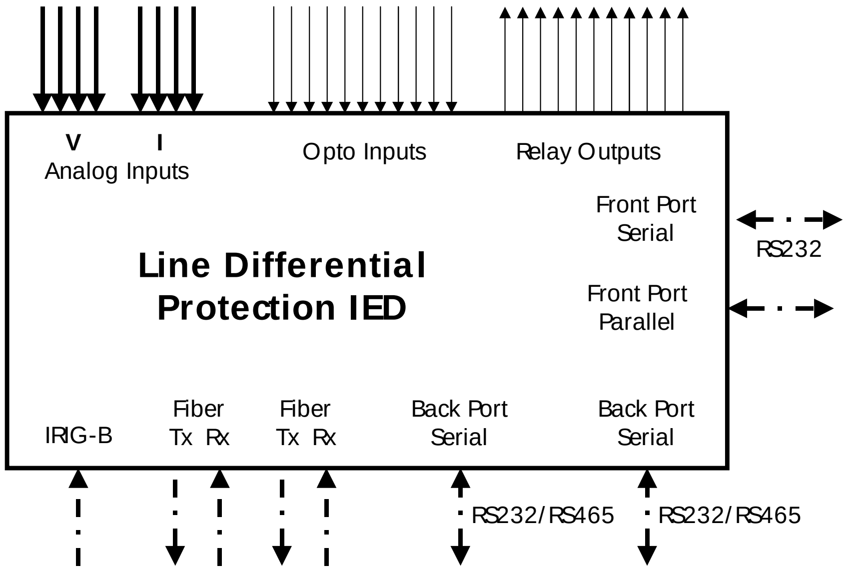


Figure 10.4: Line differential protection IED with multiple communication ports

10.4.2 Distance protection communications based schemes

One of the disadvantages of pure line differential relays is that they don't provide any form of remote backup protection and they can not function in the case of communications failure. This is what makes distance protection relay very popular for the protection of transmission line.

One disadvantage of distance relays is that they do not provide instantaneous tripping of both ends of the line that might be required to ensure system stability and to apply high-speed single pole trip and reclosing.

As a result, communications based protection schemes have been successfully used for many years. In the conventional permissive, blocking, direct tripping, etc. schemes, the outputs of distance relays are used to energize inputs of communications devices that send the signal over some carrier to the remote end of the line. The receiving communications device will close an output that will energize an input of the distance relay indicating to the scheme logic that a carrier signal has been received. Depending on the scheme logic used, the relay will trip or block.

The main problem with this kind of arrangement is the operating time of such a scheme is affected by the time of the relay outputs and the processing of the opto inputs of the relays and communications devices at both ends of the line that in some cases might not be acceptable.

Replacing the relay outputs and opto inputs with a direct communications interface between the distance relay and the communications equipment reduces costs and improves performance.

The issues for redundancy can be addressed here as well by implementing a directional comparison scheme that runs in parallel with the distance based schemes. The directional comparison scheme may use the same communications channel as the permissive or blocking distance schemes, or it can use a separate channel (Fig. 10.5) in order to improve the reliability of the protection system.

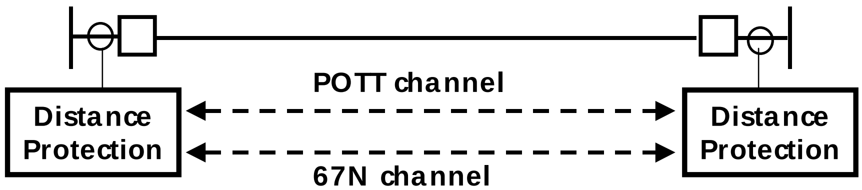


Figure 10.5: Communications based distance protection system with redundant communications channels

Fig. 10.6 shows a multifunctional protection IED with a single fiber interface for the communications interface with the remote end distance relay. There are several additional communication interfaces shown on the drawing that are described in the following sections.

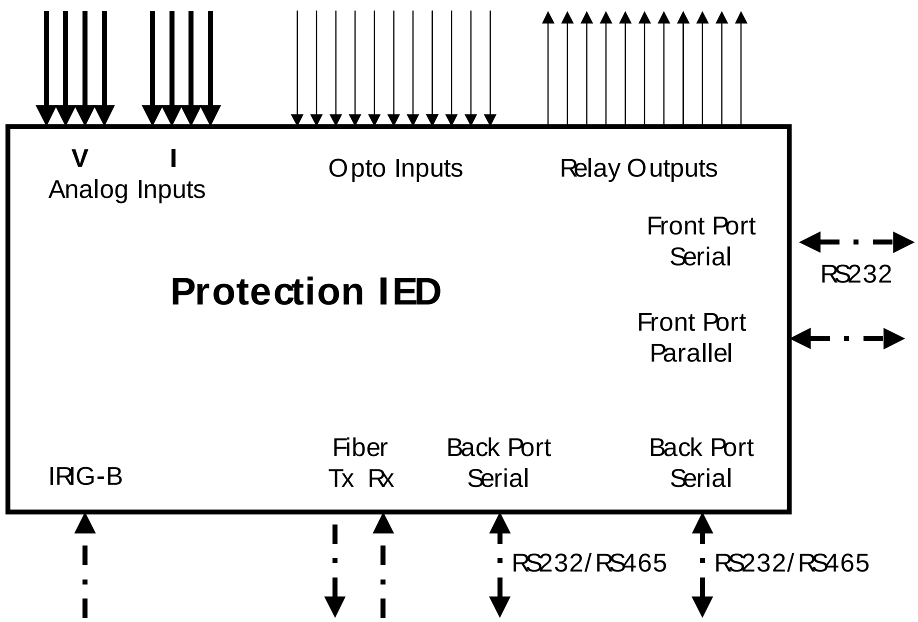


Figure 10.6: Interface of modern microprocessor based relay

10.5 Communication requirements for measurements and control

Modern microprocessor based relays have significant processing power that is used in the random cases of faults or other abnormal conditions in the power system. In order to improve the efficiency of the use of their capabilities, they perform multiple current and voltage measurements and calculate numerous system parameters.

As a result, microprocessor based relays have become the main data source for measurements in substation automation systems. The substation HMI client will require continuous polling of the individual IED's to refresh the HMI display and if necessary to provide a subset of the measurements to the SCADA master.

A very common protocol for data acquisition in the substation is DNP 3.0. In this case the IED's will typically be daisy-chained over a shielded twisted pair RS 485 using half-duplex communications with the substation client application running on a PC.

We have noticed that even at the early stages of communications interface of microprocessor based relays there was a serial port on the back of the relay (Fig. 10.1).

However, since that port is now busy providing measurements to the substation master, and considering the fact that DNP 3.0 does not support settings and (until recently) records extraction, there is obviously a need for a second serial port on the back of the relay (Fig. 10.4 and Fig. 10.6).

10.6 Communication requirements for time-synchronization

The need for time synchronization of the individual relays in a substation and even in the power system is very essential, because it allows the development and implementation of event recorders. This requires the synchronization of all IED's with an accuracy of typically one millisecond. Since this accuracy cannot be achieved using the commonly used communication protocols, the solution is to install in the substation a GPS receiver that will interface with the IED's through the most popular IRIG-B communications port (Fig. 10.4 and Fig. 10.6).

Time synchronization can also be achieved over the Ethernet using SNTP as defined in IEC 61850 for time stamping of relay operation events.

10.7 Communication requirements for maintenance

Modern state-of-the art multifunctional protection IED's are being continuously improved in order to meet the ever increasing requirements of their multiple users in a utility or industrial plant. This requires two different kinds of maintenance:

- Upgrade of the IED's firmware
- Maintenance testing to ensure that the IED meets the functional requirements of the application

Fig. 10.4 and Fig. 10.6 show a parallel port on the front of the protection IED that can be used for both types of maintenance. When the requirement is to upgrade the relay firmware, a laptop is connected to the parallel port and the user downloads the new firmware in the flash-memory of the relay.

In cases when the relay has to be tested, a simple monitoring devices with several LED's and audio alarm can be plugged into the parallel port in order to indicate the operation of different internal protection elements or schemes.

10.8 High-speed peer-to-peer communications requirements

The development of IEC 61850 has opened the doors for a new trend in power system protection and control – replacing the web of hard wires in the substation used to interface relay outputs with opto inputs with high-speed communications.

Replacing the current and voltage circuits with sampled analog values over fiber will lead to a copperless interface with the protection IED.

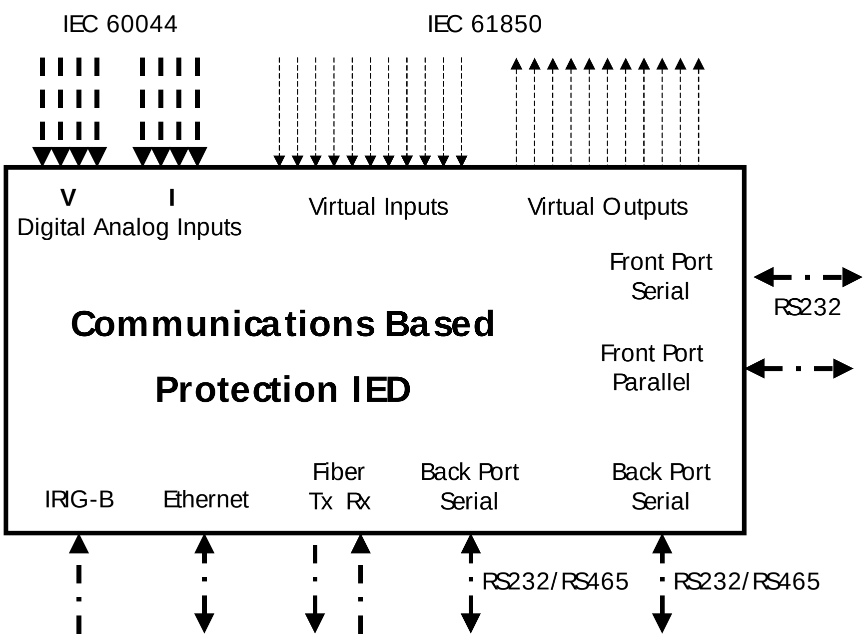


Figure 10.7: Communications based protection IED

Fig. 10.7 shows a simplified diagram of such a device. The analog and binary inputs, as well as the relay outputs are based on communications defined by the international standards above. The IEC 61850 interface can be used also for measurements data acquisition, remote control, setting changes, event, fault and disturbance records extraction and time synchronization.

The serial and front port interface is optional, if the user requires a redundant access to the substation through modem for setting changes and records extraction.

BIBLIOGRAPHY

A few books, IEEE tutorials and papers published in the literature are listed here. A comprehensive list of publications is given in the bibliography included in Reference 10.

Books and Tutorials:

1. C. R. Mason, *The Art and Science of Protective Relaying,* New York: John Wiley & Sons, 1956.
2. A. R. van C. Warrington, *Protective Relays: Their Theory and Practice*, Vol. I, New York: John Wiley and Sons, 1962.
3. Oppenheim, Alan V. and Schafer, Ronald W., Digital Signal Processing, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1975.
4. Sachdev, M.S., (Coordinator), IEEE Tutorial Course Text, Computer Relaying, New Jersey: IEEE, Publication No. 79 EH0148-7-PWR, 1979.
5. J. L. Blackburn, Protective Relaying; Principles and Applications, New York: Marcel Dekker, 1987.
6. A.G. Phadke and J. S. Thorp, Computer Relaying for Power Systems, Taunton, UK: Research Studies Press and New York: John Wiley and Sons, 1988.
7. Sachdev, M.S., (Coordinator), IEEE Tutorial Course Text, Microprocessor Relays and Protection Systems, New Jersey: IEEE, Publication No. 88EH0269-1-PWR, 1988.
8. S.H. Horowitz and A.G. Phadke, Protective System Relaying, Taunton, UK: Research Studies Press, 1992.
9. W. A. Elmore (Editor): ABB, Protective Relaying: Theory and Applications, New York: Marcel Dekker, 1994.
10. H. Ungrad, W. Winkler, and A. Wiszniewski, Protection Techniques in Electrical Energy Systems, Marcel Dekker, Inc, 1995.
11. A.T. Johns, S.K. Salman, Digital Protection for Power Systems Peter Peregrinus, Ltd, 1995.
12. Sachdev, M.S., (Coordinator), IEEE Tutorial Course Text, Advancements in Microprocessor Based Protection and Communication, New Jersey: IEEE, Publication No. 97TP120-0, 1997.
13. Ziegler, Numerical Distance Protection, Publicis Corporate Publishing, 2008.

Papers & Patents

14. Sachdev, M.S. and Baribeau, M.A., "A New Algorithm for Digit**al Impedance Rela**ys", *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-98, No. 6, November/December 1979, pp. 2232-2240.
15. Girgis A.A. and Hwang T.L.D., "Optimal Estimation of Voltage Phasors and Frequency Deviation Using Linear and Non-Linear Kalman Filtering", *IEEE Transactions on Power Apparatus and Systems*, Vol.103, No.10, 1984, pp.2943-9.
16. Lumbarsky A. and Simkins D.N., "Digital Signal Processing Apparatus for Analog Signals Having Wide Frequency and Dynamic Ranges", US patent No.777092, April 23, 1988.
17. Sachdev M.S. and Giray M.M., "Off-Nominal Frequency Measurements in Electric Power Systems", IEEE Transactions on Power Delivery, Vol.4, No.3, 1989, pp.1573-8.

18. Schweitzer, III, Edmund O. and Hou, Daqing, “Filtering for Protective Relays”, 47th Annual Georgia Tech Protective Relaying Conference, Atlanta, GA, April 20-23, 1993.

19. Terzija V.V., Djuric M.B. and Kovacevic B.D., “Voltage Phasor and Local System Frequency Estimation Using Newton-Type Algorithms”, IEEE Transactions on Power Delivery, Vol.9, No.3, 1994, pp.1368-74.

20. McLaren, P.G., Swift, G.W., Neufeld, A., Zhang, Z., Dirks, E. and Haywood, R.W., "‘Open’ Systems Relaying", IEEE Trans. Power Deliv., Vol 9, No. 3, July 1994, pp. 1316-1324

21. Kasztenny B., and Rosolowski E., “Two New Measuring Algorithms for Generator and Transformer Relaying”, IEEE Transactions on Power Delivery, Vol.13, No.4, October 1998, pp.1060-1066.

22. Das, Ratan, Benco, Joe and Downs, Cliff , “Current Based Frequency Tracking and Apparatus”, U.S. Patent No. 6,298,309 B1, October 2, 2001.

23. Sidhu, T.S.(Chair), "Bibliography of Relay Literature, 2001 IEEE Committee Report", IEEE Trans. Power Deliv., Vol. 18, No. 2, April 2003, pp. 420-429 (This paper provides reference to bibliography of relay literature since 1927).

24. Sidhu T.S., Zhang, X. and Balamourougan V., “A New Half-Cycle Phasor Estimation Algorithm”, IEEE Transactions on Power Delivery, Vol. 20-2, April 2005.

APPENDIX A

.

A.1 Least Squares Technique

Least Error Squares (LES) technique [1] is used to estimate the phasors of the fundamental and harmonic frequency components of voltages and currents. It is based on minimizing the mean-square error between the actual and assumed waveforms. The voltage and/or current waveform is modeled as a combination of the fundamental frequency component, an exponentially decaying dc component and harmonics of specified orders.

$$v(t) = V_0 e^{-\frac{t}{\tau}} + \sum_{n=1}^{N} V_n \sin(n\omega_0 t + \theta_n) \qquad \text{(A.1)}$$

where: $v(t)$ is the instantaneous value of the voltage at time t.

τ is the time constant of the decaying dc component.

N is the highest order of the harmonic component present in the signal.

ω_0 is the fundamental frequency of the system.

V_0 is the magnitude of the dc offset at $t = 0$.

V_n is the peak value of the n^{th} harmonic component.

θ_n is the phase angle of the n^{th} harmonic component.

Expressing the decaying dc component by the Taylor series expansion and retaining the first two terms, the following equation is obtained.

$$v(t) = V_0 - \left(\frac{V_0}{\tau}\right)t + \sum_{n=1}^{N} V_n \sin(n\omega_0 t + \theta_n) \qquad \text{(A.2)}$$

Assume that the voltage is composed of an exponentially decaying dc component, the fundamental frequency component and components of the second, third, fourth and fifth harmonics. For $t = t_1$, Equation A.2 can be expressed as

$$\begin{aligned} v(t_1) = V_0 - \left(\frac{V_0}{\tau}\right)t_1 + V_1 \sin(\omega_0 t_1 + \theta_1) + V_2 \sin(2\omega_0 t_1 + \theta_2) \\ + V_3 \sin(3\omega_0 t_1 + \theta_3) + V_4 \sin(4\omega_0 t_1 + \theta_4) + V_5 \sin(5\omega_0 t_1 + \theta_5) \end{aligned} \qquad \text{(A.3)}$$

Using trigonometric identities, Equation A.3 can be rewritten as

$$
\begin{aligned}
v(t_1) = V_0 - \left(V_0\big/\tau\right)t_1 &+ (V_1\cos\theta_1)\sin\omega_0 t_1 + (V_1\sin\theta_1)\cos\omega_0 t_1 \\
&+ (V_2\cos\theta_2)\sin 2\omega_0 t_1 + (V_2\sin\theta_2)\cos 2\omega_0 t_1 \\
&+ (V_3\cos\theta_3)\sin 3\omega_0 t_1 + (V_3\sin\theta_3)\cos 3\omega_0 t_1 \\
&+ (V_4\cos\theta_4)\sin 4\omega_0 t_1 + (V_4\sin\theta_4)\cos 4\omega_0 t_1 \\
&+ (V_5\cos\theta_5)\sin 5\omega_0 t_1 + (V_5\sin\theta_5)\cos 5\omega_0 t_1
\end{aligned} \quad . \tag{A.4}
$$

Proceeding in this manner, Equation A.4 can be expressed as

$$
\begin{aligned}
x(t_1) = {} & a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + a_{14}x_4 + a_{15}x_5 + a_{16}x_6 + a_{17}x_7 \\
& + a_{18}x_8 + a_{19}x_9 + a_{110}x_{10} + a_{111}x_{11} + a_{112}x_{12} + \dots\dots\dots \\
& + a_{1(2N+1)}x_{(2N+1)} + a_{1(2N+2)}x_{(2N+2)}
\end{aligned} \tag{A.5}
$$

where: $x_1 = V_0$, $x_2 = -V_0/\tau$, $x_3 = V_1\cos\theta_1$, $x_4 = V_1\sin\theta_1$,
$x_5 = V_2\cos\theta_2$, $x_6 = V_2\sin\theta_2$, $x_7 = V_3\cos\theta_3$, $x_8 = V_3\sin\theta_3$,
$x_9 = V_4\cos\theta_4$, $x_{10} = V_4\sin\theta_4$, $x_{11} = V_5\cos\theta_5$, $x_{12} = V_5\sin\theta_5$,
................................$x_{(2N+1)} = V_N\cos\theta_N$, $x_{(2N+2)} = V_N\sin\theta_N$
$a_{11} = 1$, $a_{12} = t_1$, $a_{13} = \sin(\omega_0 t_1)$, $a_{14} = \cos(\omega_0 t_1)$,
$a_{15} = \sin(2\omega_0 t_1)$, $a_{16} = \cos(2\omega_0 t_1)$, $a_{17} = \sin(3\omega_0 t_1)$, $a_{18} = \cos(3\omega_0 t_1)$,
$a_{19} = \sin(4\omega_0 t_1)$, $a_{110} = \cos(4\omega_0 t_1)$, $a_{111} = \sin(5\omega_0 t_1)$, $a_{112} = \cos(5\omega_0 t_1)$,
…………………………………… $a_{1(2N+1)} = \sin(N\omega_0 t_1)$, $a_{1(2N+2)} = \cos(N\omega_0 t_1)$.

Considering that the signal is sampled at intervals of Δt seconds, Equation A.6 can be obtained by substituting $t_1 = m\Delta t$ in Equation A.5.

$$
\begin{aligned}
x(m\Delta t) = {} & a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + a_{m4}x_4 + a_{m5}x_5 + a_{m6}x_6 + a_{m7}x_7 \\
& + a_{m8}x_8 + a_{m9}x_9 + a_{m10}x_{10} + a_{m11}x_{11} + a_{m12}x_{12} + a_{m13}x_{13} + \dots\dots \\
& + a_{m(2N+1)}x_{(2N+1)} + a_{m(2N+2)}x_{(2N+2)}
\end{aligned} \tag{A.6}
$$

where: $\Delta t = \dfrac{1}{f_s}$, (A.7)

f_s is the sampling frequency.
m is the sample number.

The a-coefficients are now redefined as follows:

$a_{m1} = 1$, $a_{m2} = m\Delta t$, $a_{m3} = \sin(\omega_0 m\Delta t)$, $a_{m4} = \cos(\omega_0 m\Delta t)$, $a_{m5} = \sin(2\omega_0 m\Delta t)$,
$a_{m5} = \sin(2\omega_0 m\Delta t)$, $a_{m6} = \cos(2\omega_0 m\Delta t)$, $a_{m7} = \sin(3\omega_0 m\Delta t)$, $a_{m8} = \cos(3\omega_0 m\Delta t)$,
$a_{m9} = \sin(4\omega_0 m\Delta t)$, $a_{m10} = \cos(4\omega_0 m\Delta t)$, $a_{m11} = \sin(5\omega_0 m\Delta t)$, $a_{m12} = \cos(5\omega_0 m\Delta t)$,
…………………………………….. $a_{m(2N+1)} = \sin(N\omega_0 m\Delta t)$, $a_{m(2N+2)} = \cos(N\omega_0 m\Delta t)$.

A total of [(2N+2)+1] equations, similar to Equation A.6, can be formed using (2N+3) consecutive samples. These can be written as

$$\underset{(2N+3)\times(2N+2)}{[A]}\ \underset{(2N+2)\times 1}{[X]} = \underset{(2N+3)\times 1}{[v]} \qquad \text{(A.8)}$$

In this equation, N [= (P-2) ÷ 2] is the highest order of the harmonic component present in the modeled signal and P is the number of samples per cycle.

The least error squares estimate of $[X]$ is given by the following equation.

$$\begin{aligned}[X] &= \left[[A]^T [A]\right]^{-1} [A]^T [v] \\ &= [A]^{\dagger} [v] \end{aligned} \qquad \text{(A.9)}$$

In this equation $[A]^{\dagger}$ is the left pseudo-inverse of $[A]$.

APPENDIX B

B.1 Line Protection Algorithms

As shown in Section 5.2, the voltage at the relay location can be expressed as a function of the current in the circuit as follows

$$v(t) = Ri(t) + L\frac{di}{dt} \qquad \text{(B.1)}$$

Integrating both sides of this equation from time t_0 to t_1 and t_1 to t_2 provides the following two equations.

$$\int_{t_0}^{t_1} v(t)dt = R\int_{t_0}^{t_1} i(t)dt + L\left(i(t_1) - i(t_0)\right) \qquad \text{(B.2)}$$

$$\int_{t_1}^{t_2} v(t)dt = R\int_{t_1}^{t_2} i(t)dt + L\left(i(t_2) - i(t_1)\right) \qquad \text{(B.3)}$$

Using the trapezoidal rule for integration and assuming that the time interval between consecutive samples is small, the integrals from sample *k* to sample *k+1* and from sample *k+1* to sample *k+2* are given by

$$\int_{t_k}^{t_{k+1}} v(t)dt = \frac{\Delta T}{2}\left[v_k + v_{k+1}\right] \qquad \text{(B.4)}$$

$$\int_{t_k}^{t_{k+1}} i(t)dt = \frac{\Delta T}{2}\left[i_k + i_{k+1}\right] \qquad \text{(B.5)}$$

$$\int_{t_{k+1}}^{t_{k+2}} v(t)dt = \frac{\Delta T}{2}\left[v_{k+1} + v_{k+2}\right] \qquad \text{(B.6)}$$

$$\int_{t_{k+1}}^{t_{k+2}} i(t)dt = \frac{\Delta T}{2}\left[i_{k+1} + i_{k+2}\right] \qquad \text{(B.7)}$$

The subscripts in these equations correspond to the sample numbers of the voltages and currents. Substituting these equations in Equations B.2 and B.3, rearranging the resulting equations, and expressing them in the matrix form, provides

$$\begin{bmatrix} \frac{\Delta T}{2}(i_{k+1}+i_k) & (i_{k+1}-i_k) \\ \frac{\Delta T}{2}(i_{k+2}+i_{k+1}) & (i_{k+2}-i_{k+1}) \end{bmatrix} \begin{bmatrix} R \\ L \end{bmatrix} = \begin{bmatrix} \frac{\Delta T}{2}(v_{k+1}+v_k) \\ \frac{\Delta T}{2}(v_{k+2}+v_{k+1}) \end{bmatrix} \tag{B.8}$$

The resistance and inductance, calculated by using the Cramer's rule, are provided by the following equations

$$R = \left[\frac{(v_{k+1}+v_k)(i_{k+2}-i_{k+1}) - (v_{k+2}+v_{k+1})(i_{k+1}-i_k)}{2\left(i_k i_{k+2} - i_{k+1}^2\right)} \right] \tag{B.9}$$

$$L = \frac{\Delta T}{2} \left[\frac{(v_{k+2}+v_{k+1})(i_{k+1}+i_k) - (v_{k+1}+v_k)(i_{k+2}+i_{k+1})}{2\left(i_k i_{k+2} - i_{k+1}^2\right)} \right] \tag{B.10}$$

B.2 Differential Equation Algorithms for Transformer Protection

As shown in Section 5, the primary voltage, v_1 may be expressed as a function of the current i_1 in the primary winding, the resistance r_1 and leakage inductance l_1 of the primary winding and the mutual flux linkages Λ as follows.

$$v_1 = r_1 i_1 + l_1 \frac{di_1}{dt} + \frac{d\Lambda}{dt} \tag{B.11}$$

Similarly, the voltage v_2 can be expressed as a function of the current, i_2, in the secondary winding and the resistance r_2 and the leakage inductance l_2 of the secondary winding and the mutual flux linkages as follows.

$$v_2 = -r_2 i_2 - l_2 \frac{di_2}{dt} + \frac{d\Lambda}{dt} \tag{B.12}$$

Equations B.11 and B.12 can be combined to eliminate the mutual flux linkages; this provides the following equation

$$v_1 = r_1 i_1 + l_1 \frac{di_1}{dt} + v_2 + r_2 i_2 + l_2 \frac{di_2}{dt} \tag{B.13}$$

Integrating both sides of this equation from t_1 to t_2 provides

$$\int_{t_1}^{t_2} v_1 dt = \left[r_1 \int_{t_1}^{t_2} i_1 dt \right] + l_1 \{ i_1(t_2) - i_1(t_1) \} + \left(\int_{t_1}^{t_2} v_2 dt \right) + \left(r_2 \int_{t_1}^{t_2} i_2 dt \right) + l_2 \{ i_2(t_2) - i_2(t_1) \} \qquad \text{(B.14)}$$

If the time from t_1 to t_2 is one sampling interval, ΔT, and the samples taken at time t_1 are identified by the subscript k, the samples taken at time t_2 are identified by the subscript $k+1$ and the trapezoidal rule is used to integrate the terms in Equation B.14, the following equation is obtained

$$\begin{aligned} \frac{\Delta T}{2}\left(v_{1(k)} + v_{1(k+1)}\right) = \frac{\Delta T}{2} r_1 \left(i_{1(k)} + i_{1(k+1)}\right) + l_1 \left(i_{1(k+1)} - i_{1(k)}\right) + \frac{\Delta T}{2}\left(v_{2(k)} + v_{2(k+1)}\right) \\ + \frac{\Delta T}{2} r_2 \left(i_{2(k)} + i_{2(k+1)}\right) + l_2 \left(i_{2(k+1)} - i_{2(k)}\right) \end{aligned} \qquad \text{(B.15)}$$

This equation expresses the integration of the primary voltage as a function of the secondary voltage, currents in the primary and secondary windings and the transformer parameters. Because the transformer parameters are known, and the primary and secondary voltages and currents are sampled and quantized, the left and right-hand sides of Equation B.15 can be calculated at every sampling instant. During normal operation, external faults and magnetizing inrush, the calculated values of the left and right-hand sides of Equation B.15 would be equal for all practical purposes but will not be equal during faults in the protection zone of the transformer.

B.2.1 Application to three phase transformers

The technique described in the previous section can be extended to protect banks of three single phase transformers and wye–wye connected transformers.

In this case, the currents in the delta winding have two components, the circulating currents and the non-circulating currents. When the currents in the delta winding of the transformer can be measured, the technique described in the previous section can be applied. If it is possible to measure the currents in the lines leading into the delta winding only, the non-circulating components of the currents in delta windings can be obtained by using the following equations.

$$i_A = \frac{1}{3}\left(i_{LA} - i_{LB}\right); \qquad i_B = \frac{1}{3}\left(i_{LB} - i_{LC}\right); \qquad i_C = \frac{1}{3}\left(i_{LC} - i_{LA}\right); \qquad \text{(B.16)}$$

Now following the procedure described earlier to the phase A-a of the transformer the following equation can be obtained.

$$\frac{\Delta T}{2}\left(v_{A(k)}+v_{A(k+1)}\right)=\left(\frac{\Delta T}{2}r_A+l_A\right)i_{A(k)}+\left(\frac{\Delta T}{2}r_A-l_A\right)i_{A(k+1)}+\left(\frac{\Delta T}{2}r_A+l_A\right)i_{p(k)}+ \left(\frac{\Delta T}{2}r_A-l_A\right)\left(+i_{p(k+1)}\right)+\frac{\Delta T}{2}\left(v_{a(k)}+v_{a(k+1)}\right)+\frac{\Delta T}{2}r_a\left(i_{a(k)}+i_{a(k+1)}\right)+l_a\left(i_{a(k+1)}-i_{a(k)}\right) \quad \text{(B.17)}$$

Following a similar procedure, the following equations can be obtained for the other two phases.

$$\frac{\Delta T}{2}\left(v_{B(k)}+v_{B(k+1)}\right)=\left(\frac{\Delta T}{2}r_B+l_B\right)i_{B(k)}+\left(\frac{\Delta T}{2}r_B-l_B\right)i_{B(k+1)}+\left(\frac{\Delta T}{2}r_B+l_B\right)i_{p(k)}+ \left(\frac{\Delta T}{2}r_B-l_B\right)\left(+i_{p(k+1)}\right)+\frac{\Delta T}{2}\left(v_{b(k)}+v_{b(k+1)}\right)+\frac{\Delta T}{2}r_b\left(i_{b(k)}+i_{b(k+1)}\right)+l_b\left(i_{b(k+1)}-i_{b(k)}\right) \quad \text{(B.18)}$$

$$\frac{\Delta T}{2}\left(v_{C(k)}+v_{C(k+1)}\right)=\left(\frac{\Delta T}{2}r_C+l_C\right)i_{C(k)}+\left(\frac{\Delta T}{2}r_C-l_C\right)i_{C(k+1)}+\left(\frac{\Delta T}{2}r_C+l_C\right)i_{p(k)}+ \left(\frac{\Delta T}{2}r_C-l_C\right)\left(+i_{p(k+1)}\right)+\frac{\Delta T}{2}\left(v_{c(k)}+v_{c(k+1)}\right)+\frac{\Delta T}{2}r_c\left(i_{c(k)}+i_{c(k+1)}\right)+l_c\left(i_{c(k+1)}-i_{c(k)}\right) \quad \text{(B.19)}$$

On taking samples of terminal voltages and line currents and quantizing them, the currents in the delta windings can be calculated using Equation B.16. From the quantized values of v_A, v_a and i_a and the calculated value of i_A, the following function of i_p can be calculated.

$$\left(\frac{\Delta T}{2}r_A+l_A\right)i_{p(k)}+\left(\frac{\Delta T}{2}r_A-l_A\right)\left(+i_{p(k+1)}\right)=\frac{\Delta T}{2}\left(v_{A(k)}+v_{A(k+1)}\right)-\left(\frac{\Delta T}{2}r_A+l_A\right)i_{A(k)} -\left(\frac{\Delta T}{2}r_A-l_A\right)i_{A(k+1)}-\frac{\Delta T}{2}\left(v_{a(k)}+v_{a(k+1)}\right)-\frac{\Delta T}{2}r_a\left(i_{a(k)}+i_{a(k+1)}\right)-l_a\left(i_{a(k+1)}-i_{a(k)}\right) \quad \text{(B.20)}$$

It can be shown that

$$\left(\frac{\Delta T}{2}r_B+l_B\right)i_{p(k)}+\left(\frac{\Delta T}{2}r_B-l_B\right)\left(+i_{p(k+1)}\right)\cong\left[\begin{array}{l}\left(\frac{\Delta T}{2}r_A+l_A\right)i_{p(k)}+\\ \left(\frac{\Delta T}{2}r_A-l_A\right)\left(+i_{p(k+1)}\right)\end{array}\right]\times\frac{2l_B+\Delta T r_B}{2l_A+\Delta T r_A} \quad \text{(B.21)}$$

Having estimated the function defined by this equation, it becomes possible to calculate the left and right hand sides of Equation B.21. This equation and a similar Equation for phase C-c can be used to determine if the transformer is experiencing a fault or not.

The procedure can be implemented by calculating the absolute values of the right-hand and left-hand sides of these equations and determining if the absolute value of the difference exceeds a threshold. If it does, a counter can be incremented. When the counter exceeds a specified ceiling, a trip command can be issued. If the absolute value of the difference is less than threshold, the counter can be decremented. The counter reaches a minimum threshold, it can be concluded that the transformer is not experiencing a fault.