

Pulse Propagation in Multiaccess Wireline Channels

by

Andrew Selasi Agbemenu

BSc. Electrical/Electronic Engineering (Hons.)

A Thesis submitted to the Department of Electrical/Electronic Engineering,

Kwame Nkrumah University of Science and Technology

in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Telecommunications Engineering

Faculty of Electrical and Computer Engineering

College of Engineering

September 2011

Dedicated to

To Lovelace and Katherine for the encouragement

KNUST



Abstract

In telecommunication, a lot of measurements and models are performed in the time domain. Time domain analysis are usually complex and difficult difference equations are employed.

Frequency Domain Models reduce the complexity of some of the mathematics to simpler arithmetic and convolution where necessary.

In this thesis, Frequency Domain Modeling is used to come up with two models; Voltage Propagation Model (VPM) and the Current Propagation Models (IPM). The graph analysis is done in the time domain.

Python, a powerful array manipulation programming language with third party scientific libraries: scipy, numpy and matplotlib, is used in the programming of the model. Since most models do not integrate GUI for easy testing, a GUI is also incorporated using Nokia's Qt framework.

The simulation results show that both the voltage and current pulses behave in similar ways. The attenuation and distortion of the pulses increases with decrease in the pulse width (increase frequency of propagation). The attenuation of the pulses also increases with increase in the number of taps on the transmission line.

Declaration

I hereby declare that this submission is my own work towards the MSc and that, to the best of my knowledge, it contains no material previously published by another person nor material which has been accepted for the award of any other degree of the University, except where due acknowledgement has been made in the text.

.....
Student Name & ID

Signature

Date

Certified by:

.....
Supervisor(s) Name

Signature

Date

Certified by:

.....
Head of Dept. Name

Signature

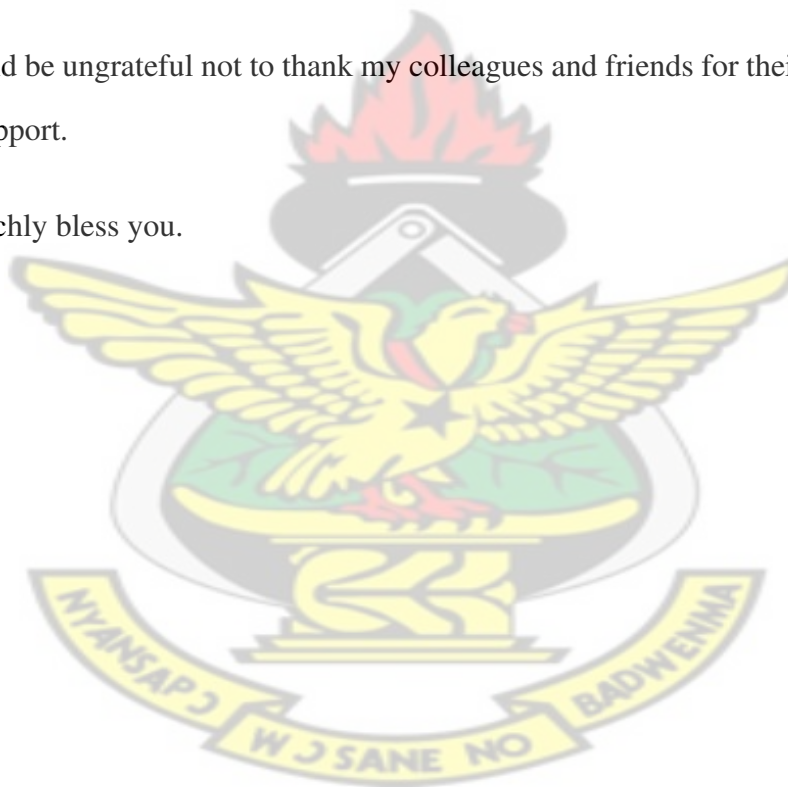
Date

Acknowledgements

First of all I thank God for guiding and directing my path to this level. I also thank Rev. Dr. J.K Oppong for supervising me and being patient with me during my candidature.

It would be ungrateful not to thank my colleagues and friends for their encouragement and support.

God richly bless you.



Contents

Abstract	iii
Declaration	iv
Acknowledgements	v
Table of Contents	ix
List of Figures	xii
1 Introduction	1
1.1 Background	1
1.2 Problem Statement	2
1.3 Project Objective	2
1.4 Motivation	3
1.5 Methodology	3
1.6 Scope	3
1.7 Limitations of Study	4
1.8 Outline	4

2	Transmission Line Theory	5
2.1	Equivalent Circuit	7
2.2	Impedance of an Ideal Transmission Line	7
2.3	Impedance of a Transmission Line With Losses	8
2.3.1	Transmission Line With Series Losses	8
2.3.2	Transmission Line With both Series and Parallel Losses	10
2.4	Transmission Line Equations	10
2.4.1	Determining the Equations	11
2.4.2	Solution to the Transmission Line Equation in Time Domain	13
2.4.3	Frequency Domain Solution	15
2.4.3.1	Solution to Differential Equation in the Frequency Domain	18
2.5	Phase Velocity and Propagation Constant	19
2.6	Pulses on a Transmission Line	19
2.6.1	Step and Pulse Response of a Transmission Line	20
2.7	Multi-access Transmission Line	23
2.7.1	Node model of multi-access transmission lines	24
2.8	Reflection on Transmission Lines	25
2.8.1	Reflection Coefficient of Nodes on a Transmission Line	26

3	Model Description	27
3.1	Model Assumptions	27
3.2	Model Description	28
3.2.1	Voltage Propagation Model (VPM)	29
3.2.1.1	VPM Equation	30
3.2.2	Current Propagation Model (IPM)	33
3.2.2.1	IPM Equation	34
3.2.3	Power Propagation Model (PPM)	36
4	Model Simulation and Results	37
4.1	Simulation tools	37
4.2	Voltage Propagation Model Flowchart	38
4.3	Current Propagation Model Flowchart	40
4.4	The GUI of the Model program	42
4.5	Simulation Results	43
4.6	Results Discussion	53
4.6.1	Voltage Pulse Model (VPM)	53
4.6.2	Current Propagation Model (IPM)	54
5	Conclusion and Recommendation	55
5.1	Conclusion	55
5.2	Recommendation	55

<i>CONTENTS</i>	ix
References	57
A Code Listings	59
A.1 Main Program Listing	59
A.2 GUI Listing	66
A.2.1 Pulse.ui Listing	66
A.2.2 GUI Python Listing	79
A.3 MPL_Widget Listing	86



List of Figures

2.1	The three transmission line modes [1]	6
2.2	Two common configurations of equivalent circuit of a transmission line with subsectional length l	7
2.3	Transmission line with distributed parameters	8
2.4	Transmission line with only series losses resulting from Fig 2.2 (b) . .	9
2.5	Distributed parameter model of a transmission line	11
2.6	Frequency domain (red) and time domain(black) representation of a pulse signal	15
2.7	Pulse obtained from a superposition of two step functions [2]	20
2.8	Step function response of transmission lines with various terminations [2]	21
2.9	Pulse function response of transmission lines with various terminations[2]	22
2.10	Conceptual view of multi-access transmission line	23
2.11	Circuit diagram of multi-access transmission line	23
2.12	Model of transmission line node	24

2.13	Basic Loaded transmission line with length L , input voltage V_s , input impedance V_s and terminated with load Z_L [3]	25
3.1	Multi-access transmission line divided into blocks	28
3.2	Block diagram of multi-access transmission line modeled.	29
3.3	Voltage Propagation Model (VPM) in the frequency domain	29
3.4	Voltage Propagation Model (VPM) tapped transmission line reflection model with 4 nodes [4]	31
3.5	Current Propagation Model (IPM) in the frequency domain	33
4.1	(a) Qt Designer and (b)Spyder Python IDE	38
4.2	Voltage Propagation Model flowchart	39
4.3	Current Propagation Model (IPM) Flowchart	41
4.4	Graphical User Interface and entering transmission line parameters and viewing pulse shapes	42
4.5	Voltage pulse shape at 100m on a 100m transmission line with 1 load in the middle of the transmission line with variable input current pulse width of (a) 1s (b) 1ms (c) 100 μ s (d) 10 μ s (e) 1 μ s (f) 100ns (g) 10ns (f) 1ns	45
4.6	Current pulse shape at 100m on a 100m transmission line with 1 load in the middle of the transmission line with variable input voltage pulse width of (a) 1s (b) 1ms (c) 100 μ s (d) 10 μ s (e) 1 μ s (f) 100ns (g) 10ns (f) 1ns	46

4.7	Voltage pulse shape at 100m on a 100m transmission line with no nodes on transmission line with variable input voltage pulse width of (a) 1ms (b) $1\mu\text{s}$ (c) 100ps (d) 10fs (e) $1 \times 10^{-15}\text{ms}$ (f) $1 \times 10^{-18}\text{ms}$ (g) $1 \times 10^{-19}\text{ms}$ (h) $1 \times 10^{-20}\text{ms}$	47
4.8	Voltage pulse shape at 100m on a transmission line of 100m with input pulse width of 1ms with (a) 0, 1, 4, 19, 49, and 99 nodes on the transmission line	48
4.9	Current pulse shape at 100m on a transmission line of 100m with input pulse width of 1ms with (a) 0, 1, 4, 19, 49 and 99 nodes on the transmission line	48
4.10	Voltage pulse of width (a) 1ms and (b) $1\mu\text{s}$ on a 100m long transmission line at points on the transmission line	49
4.11	Current pulse of width (a) $1\mu\text{s}$ and (b) 1ns on a 100m long transmission line at points on the transmission line	50
4.12	Pulse shapes of (a) voltage and (b) current for different characteristic inductance, L values with fixed characteristic capacitance, $C = 60.0\text{pF/m}$	51
4.13	Current pulse shapes at 100m for (a) VPM and (b) IPM for variable per unit characteristic capacitances, C with all other parameters remaining constant.	52
4.14	Voltage pulse for variable (a) characteristics per unit resistance and (b) characteristics per unit conductance	52
4.15	Current pulse for variable (a) characteristics per unit resistance, R and (b) characteristics per unit conductance, G	53

Chapter 1

Introduction

1.1 Background

Modeling of transmission lines has regained interest in the context of data communications [5]. In order to accurately evaluate the available throughput and bit-error rate (BER), an accurate model of the transmission line is required to be developed to aid designers use the right number of nodes, length and type of transmission line.

Wire-line channels are virtually present in every component that carries a signal from one place to the other. The physical layer of practically every Ethernet system is a wireline channel. Though fibre optics has a higher data rate, it is very expensive to implement and will not be replacing wireline channels anytime soon.

With the advent of Power Line Communication (PLC), it has also become increasingly important to study transmission lines. Power lines can be seen as multi-access wireline channels as they have a lot of junction points going into homes. In the homes, there are also a lot of junction points from the sockets on the walls and the light bulbs.

In a typical data communication network, repeaters are placed at some points where the signals are thought to attenuate to the minimum readable point. Proper transmission line models help study these effects so as to identify more optimal repeater points.

There are various transmission line models which are either Time Domain Models or Frequency Domain Models. Time Domain Models generally look at how the signal changes with time while Frequency Domain Models look at how the signals change with changing frequencies.

1.2 Problem Statement

Transmission lines have been studied extensively in the past. In recent times the interest in electric transmission lines in the macro form has dwindled in favour of optic fibres. Owing to the lack of adequate wireline transport infrastructure, wireless Internet technologies have been on the increase in most African countries.

Wireless technologies have a limited reach especially in areas of rich flora (eg. KNUST staff residence) because of greater absorption and dispersion of the signals. This problem can be solved by using the existing electrical wiring installation.

Most transmission line models have concentrated mainly on the signal at the end of the line, and thus it will be difficult for a designer to tell the optimum nodes to use. Modeling a transmission line to observe the signal (pulse) at every point on the line makes it easier for the designer to use the optimum nodes as access points.

1.3 Project Objective

The objective of the thesis are:

1. To design a mathematical model of a multi-access transmission line that allow one to determine the voltage and current pulse shapes at any point on the transmission line.
2. Program the designed model using an appropriate programming language (Python)

3. To develop an interactive Graphical User Interface (GUI) as a frontend to the model.

1.4 Motivation

The primary motivation behind the project reported in this document is based on the work reported in [6] in which a pulse is generated from one end of a transmission line and observed on the terminating end. This work however, looks at the pulse shape at every point on the transmission line.

With increasing interest in Power Line Communications (PLC) [7], shows that a power transmission line can be considered as a multi-access channel. This work therefore makes the power transmission line model easier to use with a simple interactive user interface.

1.5 Methodology

The methodology used in this thesis involves review of transmission line theory and models. Models are put together to form composite models for multi-access transmission lines which are used for analysis.

1.6 Scope

Firstly, this work looks at studying general transmission line models. These models are then put together to develop two mathematical models : Voltage Propagation Model (VPM) and Current Propagation Model (IPM)

A free and open source programming language, Python, is to be used in programming the model. The GUI is to be developed using Qt Designer from Nokia.

1.7 Limitations of Study

The primary limitation is the unavailability of modern research papers in this area. Most references are old and difficult to come by. Materials on multi-access models are almost non-existent.

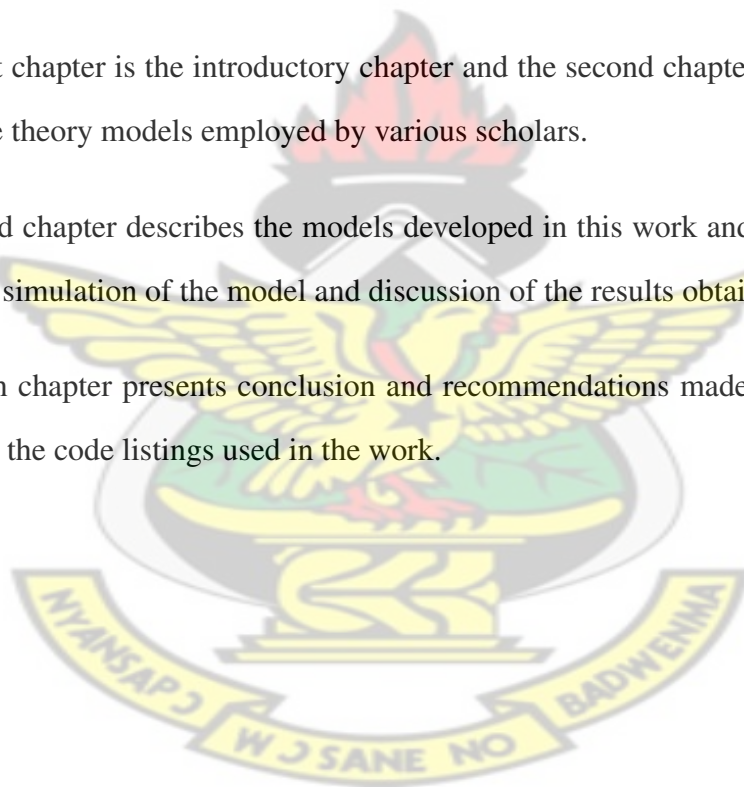
1.8 Outline

The thesis is divided into five chapters with an Appendix.

The first chapter is the introductory chapter and the second chapter reviews transmission line theory models employed by various scholars.

The third chapter describes the models developed in this work and the fourth chapter looks at simulation of the model and discussion of the results obtained.

The fifth chapter presents conclusion and recommendations made and the Appendix displays the code listings used in the work.



Chapter 2

Transmission Line Theory

A transmission line is a special device designed to carry electrical energy from one point to the other[8]. This device can either be a cable or a waveguide. Transmission lines can be studied using two approaches. The transmission line is analyzed through electromagnetism as a three-dimensional structure (waveguide) if the aim is to study the electric and magnetic fields propagating through the line where the analysis is based on the Maxwell's equations [9]. However, if the interest is the propagation of signal or energy, the analysis is simplified by discussing the waveguide as a one-dimensional mathematical model [9]. There are three common guiding structures that are used: [10]

- Stripline: parallel-plate transmission line, where two parallel conducting plates are separated by a dielectric slab of uniform thickness
- Two-wire transmission line: a pair of parallel transmission conducting wires separated by a uniform distance (parallel) or twisted together (twisted pair).
- Coaxial transmission line: an inner conductor and an outer coaxial conducting sheath separated by a dielectric.

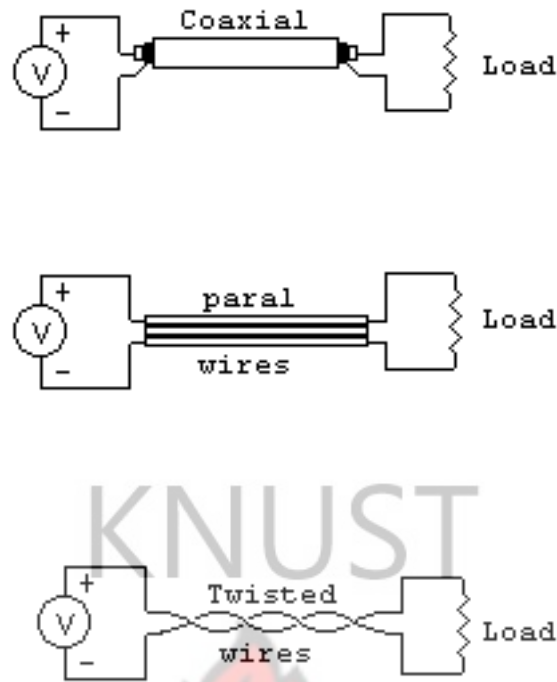


Figure 2.1: The three transmission line modes [1]

The structure and other physical factors affect the operation of a transmission line. The parameters affected are shown in Table 2.1. The Table shows that the geometry of the transmission line affects every single parameter (i.e. R , L , C and G) but loss tangent affects only the conductance, G .

Table 2.1: Transmission line parameters and factors with black dots showing affected parameters [1]

Property	R	L	C	G
Conductance	●	○	○	●
Geometry	●	●	●	●
loss tangent	○	○	○	●
proximity effect	●	●	○	○
permeability μ	○	●	○	○
permittivity ϵ	○	○	●	○
radiation	●	●	○	○
skin effect	●	●	○	○

Legend : ● Affected ○ Unaffected

2.1 Equivalent Circuit

A transmission line can be approximated by a distributed-parameter network with the circuit parameters distributed throughout the line. The distributed parameters are resistance per unit length (R), inductance per unit length (L), capacitance per unit length (C) and conductance per unit length (G). These parameters are connected together to form an equivalent circuit. There are various configurations possible, two common configurations are shown in Figure 2.2. Figure 2.2(b) is used in this work.

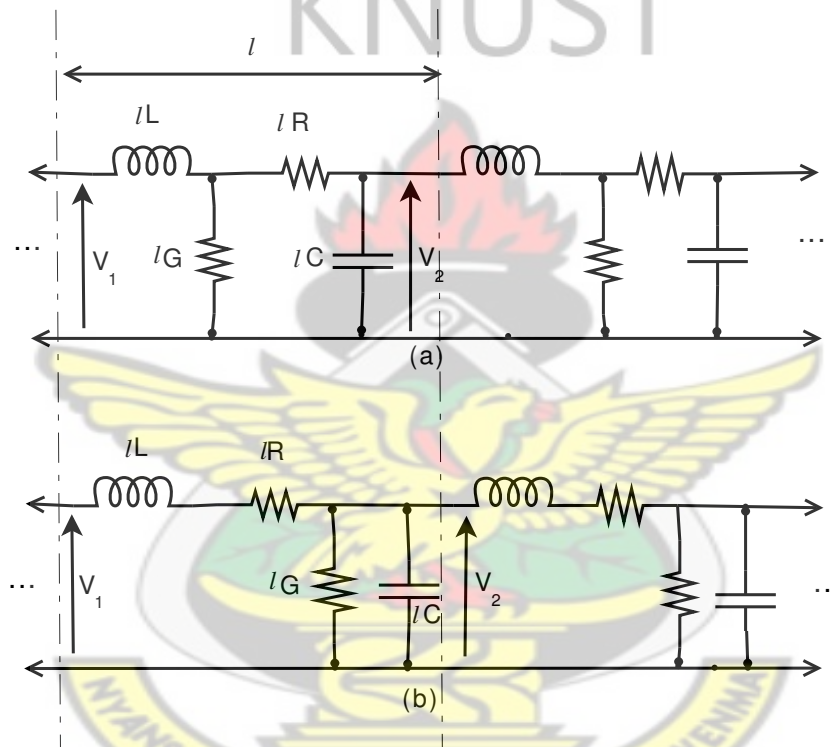


Figure 2.2: Two common configurations of equivalent circuit of a transmission line with subsectional length l

2.2 Impedance of an Ideal Transmission Line

If a transmission line has no losses, it is considered to be a repeated array of small inductors and capacitors in a ladder network [2]. This makes the resistance and conductance in Figure 2.3 negligible and can be approximated as $R = G = 0$. The input

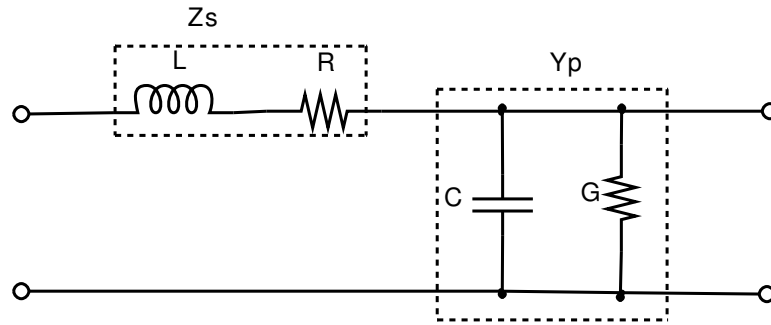


Figure 2.3: Transmission line with distributed parameters

impedance, Z_1 is given by (2.1) [2]. This equation contains complex terms that can be eliminated by making l infinitesimally small. The equation then simplifies to (2.2).

$$Z_1^2 - j\omega L l Z_1 = \frac{L}{C} \quad (2.1)$$

where,

$$Z_1 = \sqrt{\frac{L}{C}} \bigg|_{l \rightarrow 0} \quad (2.2)$$

In practical transmission line design, the series resistance of the line cannot be ignored but L and C influence the fundamental behaviour of pulses through them.

2.3 Impedance of a Transmission Line With Losses

In practical designs, the transmission lines have both series and parallel resistances. These generally vary depending on the physical design of the line and the frequency of signal propagation. Transmission line losses can either be only series losses or both series and parallel losses.

2.3.1 Transmission Line With Series Losses

When the series resistance is present, Figure 2.2(b) becomes Figure 2.4.

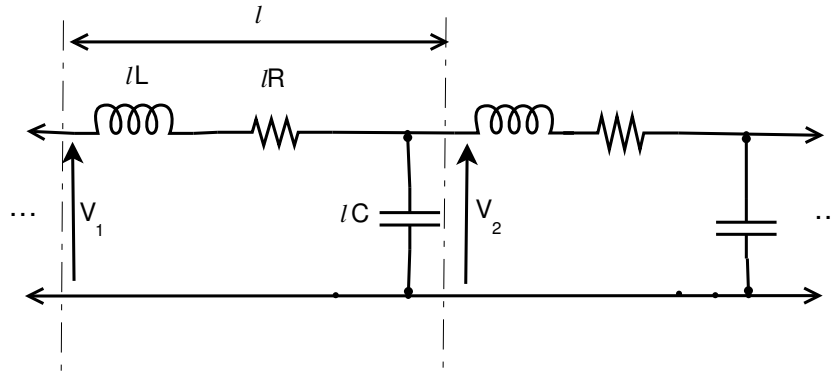


Figure 2.4: Transmission line with only series losses resulting from Fig 2.2 (b)

The characteristic impedance, Z_0 for infinitesimal values of l becomes,

$$Z_0 = \sqrt{\frac{R + j\omega L}{j\omega C}} \quad (2.3)$$

To study the effect of the series losses on the characteristic impedance, the equation can be rewritten as

$$Z_0 = \left[\frac{L}{C} \left(1 - j \frac{R}{\omega L} \right) \right]^{\frac{1}{2}} \quad (2.4)$$

The equation is then expanded using the identity $(1 \pm v)^{\frac{1}{2}} = 1 \pm \frac{v}{2}$ ¹ for small v if the losses are small. $\frac{R}{\omega L}$ small and (2.4) is approximated to

$$Z_0 = \sqrt{\frac{L}{C}} \left(1 - j \frac{R}{2\omega L} \right) \quad (2.5)$$

Since $\frac{R}{\omega L}$ is much smaller than unity, it is apparent that small series losses have little effect on the characteristic impedance of the line [2].

¹A mathematical identity which approximates the higher powers of v to 0 because of their negligible values.

2.3.2 Transmission Line With both Series and Parallel Losses

The characteristic impedance of the transmission line changes when the parallel losses are significant. Figure 2.2 (b) represents an infinitesimal section of a line with both series and parallel losses. The characteristic impedance, Z_0 becomes

$$Z_0 = \sqrt{\frac{R + j\omega L}{G + j\omega C}} \quad (2.6)$$

Now using Figure 2.3, Z_0 becomes

$$Z_0 = \sqrt{\frac{Z_s}{Y_p}} \quad (2.7)$$

2.4 Transmission Line Equations

To determine how voltage and current are affected by the various impedances as they propagate through all the segments of the line, equations relating to them and the distributed parameters have to be determined. Figure 2.5 shows the distributed parameter model with different nomenclature from the previous sections. The variables in the figure are defined as:

- Δx is an infinitesimal section of the transmission line.
- R, L, G, C are the characteristic resistance, inductance, conductance and capacitance of the transmission line.
- $i(x, t)$ and $v(x, t)$ are the current and voltage at any time, t and a distance, x along the transmission line.

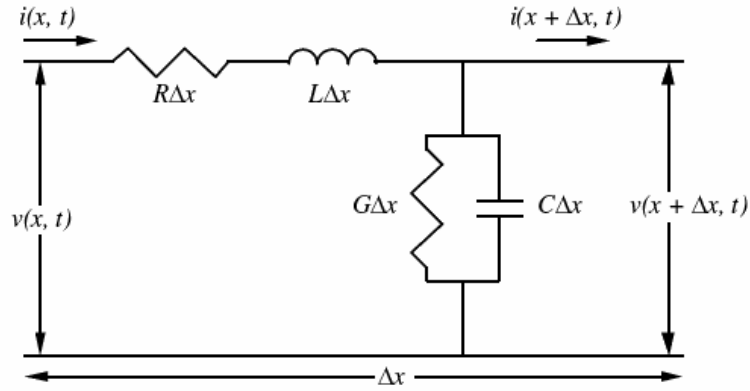


Figure 2.5: Distributed parameter model of a transmission line

2.4.1 Determining the Equations

From the characteristic equations for inductors and capacitors:

$$v_L = L \frac{\partial i_L}{\partial t} \quad (2.8)$$

and

$$i_C = C \frac{\partial v_C}{\partial t} \quad (2.9)$$

now applying Kirchhoff's Voltage Law (KVL)² to Figure 2.5 we obtain

$$v(x, t) = R\Delta x \cdot i(x, t) + L\Delta x \cdot \frac{\partial i}{\partial t}(x, t) + v(x + \Delta x, t) \quad (2.10)$$

Now,

$$\Delta v = v(x, t) - v(x + \Delta x, t) = R\Delta x \cdot i(x, t) + L\Delta x \cdot \frac{\partial i}{\partial t}(x, t) \quad (2.11)$$

²KVL states that the sum of all voltages around a closed loop is equal to zero.

Dividing (2.11) by Δx gives:

$$\frac{\Delta v}{\Delta x} = Ri(x, t) + L \frac{\partial i}{\partial t}(x, t) \quad (2.12)$$

which is simplified to

$$\frac{\partial v}{\partial x} = Ri + L \frac{\partial i}{\partial t} \quad (2.13)$$

now applying Kirchhoff's Current Law (KCL)³ to Figure 2.5 gives:

$$i(x, t) = G \Delta x \cdot v(x + \Delta x, t) + C \Delta x \frac{\partial v}{\partial t}(x + \Delta x, t) + i(x + \Delta x, t) \quad (2.14)$$

Then

$$\Delta i = i(x, t) - i(x + \Delta x, t) = G \Delta x \cdot v(x + \Delta x, t) + C \Delta x \frac{\partial v}{\partial t}(x + \Delta x, t) \quad (2.15)$$

Now dividing through by Δx yields

$$\frac{\Delta i}{\Delta x} = Gv(x + \Delta x, t) + C \frac{\partial v}{\partial t}(x + \Delta x, t) \quad (2.16)$$

This equation then simplifies to:

$$\frac{\partial i}{\partial x} = Gv + C \frac{\partial v}{\partial t} \quad (2.17)$$

Combining (2.13) and (2.17), we obtain:

³KCL simply states that the algebraic sum of the current at a node (junction) in an electrical circuit is zero.

$$\frac{\partial^2 v}{\partial x^2} = RGv + (RC + LG)\frac{\partial v}{\partial t} + LC\frac{\partial^2 v}{\partial t^2} \quad (2.18)$$

$$\frac{\partial^2 i}{\partial x^2} = RGi + (RC + LG)\frac{\partial i}{\partial t} + LC\frac{\partial^2 i}{\partial t^2} \quad (2.19)$$

Equations (2.18) and (2.19) are called *The Transmission Line Equations* [2, 1]. The first is the transmission line equation for *Voltage* and the second is the transmission line equation for *Current*.

2.4.2 Solution to the Transmission Line Equation in Time Domain

In solving (2.18), a solution is guessed for the voltage function in the form:

$$v(t) = e^{j\omega t} e^{-\gamma x} \quad (2.20)$$

where $e^{j\omega t}$ ⁴ is a unity amplitude sinusoidal wave and $e^{-\gamma x}$ (with γ being a complex variable) is a sinusoidal wave whose amplitude decays exponentially with distance.

Now let

$$\gamma = \alpha + j\beta \quad (2.21)$$

where

γ is the propagation constant

α is the attenuation coefficient in Nepers per meter and

β is the phase shift coefficient in radians per meter.

⁴Euler's Identity: $e^{j\omega t} = \cos(\omega t) + j \sin(\omega t)$

Now substituting (2.21) into (2.20) we obtain

$$v(t) = e^{j\omega t} e^{-(\alpha+j\beta)x} \quad (2.22)$$

Now substituting (2.22) into (2.18) yields

$$\begin{aligned} \frac{\partial^2}{\partial x^2} \left[e^{j\omega t} e^{-(\alpha+j\beta)x} \right] &= RG \left[e^{j\omega t} e^{-(\alpha+j\beta)x} \right] + (RC + LG) \frac{\partial}{\partial t} \left[e^{j\omega t} e^{-(\alpha+j\beta)x} \right] + \\ &LC \frac{\partial^2}{\partial t^2} \left[e^{j\omega t} e^{-(\alpha+j\beta)x} \right] \end{aligned} \quad (2.23)$$

which then simplifies to:

$$\begin{aligned} (\alpha + j\beta)^2 &= RG + (RC + LG) j\omega - LC\omega^2 \\ &= (R + j\omega L)(G + j\omega C) \end{aligned} \quad (2.24)$$

confirming [2, 11, 12]

For a lossless transmission line where $R = G = 0$, the voltage equation simplifies to

$$\frac{\partial^2 v}{\partial x^2} = LC \frac{\partial^2 v}{\partial t^2} \quad (2.25)$$

while the current equation becomes

$$\frac{\partial^2 i}{\partial x^2} = LC \frac{\partial^2 i}{\partial t^2} \quad (2.26)$$

Now substituting $R = G = 0$ into (2.24) results

$$\alpha + j\beta = j\omega\sqrt{LC} \quad (2.27)$$

which then makes

$$\alpha = 0 \quad (2.28)$$

$$\beta = \omega\sqrt{LC} \quad (2.29)$$

implying that for a lossless transmission line, signals experience only phase shift directly proportional to its frequency of propagation.

2.4.3 Frequency Domain Solution

Signal analysis is often done in the frequency domain where the effect of the system on each spectral component of the transferred signal can be easily observed. In pulse analysis in particular, it is more beneficial to perform analysis in the frequency domain because each pulse is made of a sum of infinite sinusoidal waves with related frequencies.

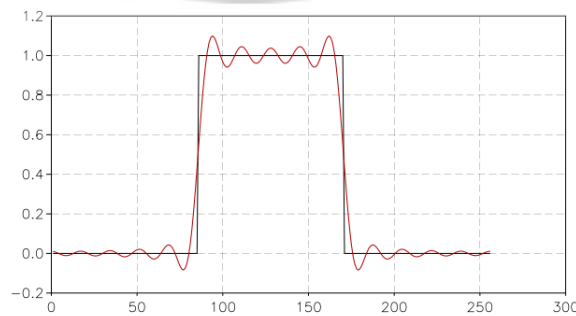


Figure 2.6: Frequency domain (red) and time domain(black) representation of a pulse signal

The frequency domain equation is determined by finding the Fourier transform of (2.18).

The Fourier transform is defined as [13, 14]

$$\mathcal{F}\{f(t)\} = F(\omega) = \int_{-\infty}^{\infty} e^{-j\omega t} f(t) dt \quad (2.30)$$

To solve this equation, we have to specify the time boundary condition and the distance boundary conditions. These conditions are:

$v \rightarrow 0$ as $x \rightarrow \infty$ and also $v \rightarrow 0$ as $t \rightarrow \infty$

The Fourier transform of (2.18) then becomes:

$$\mathcal{F}\left\{\frac{\partial^2 v}{\partial x^2}\right\} = \mathcal{F}\left\{RGv + (RC + LG)\frac{\partial v}{\partial t} + LC\frac{\partial^2 v}{\partial t^2}\right\} \quad (2.31)$$

now let

$$\mathcal{F}\{v\} = V(\omega) = V \quad (2.32)$$

The Fourier transform of the first derivative $\frac{\partial v}{\partial t}$, is

$$\mathcal{F}\left\{\frac{\partial v}{\partial t}\right\} = \int_{-\infty}^{\infty} e^{-j\omega t} \frac{\partial v}{\partial t} dt \quad (2.33)$$

Using integration by parts⁵ we obtain

⁵Integration by parts is defined: $\int u \cdot dw = uw - \int w \cdot du$ where $u = e^{-j\omega t} \rightarrow du = -j\omega e^{-j\omega t}$ and $dw = \frac{\partial v}{\partial t}$ and $w = v$

$$\begin{aligned}
\mathcal{F}\left\{\frac{\partial v}{\partial t}\right\} &= e^{-j\omega t}v\Big|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} v(-j\omega e^{-j\omega t})dt \\
&= j\omega \int_{-\infty}^{\infty} e^{-j\omega t}vdt \\
&= j\omega \mathcal{F}\{v\} \\
&= j\omega V
\end{aligned} \tag{2.34}$$

Similarly:

$$\begin{aligned}
\mathcal{F}\left\{\frac{\partial^2 v}{\partial t^2}\right\} &= (j\omega)^2 \mathcal{F}\{v\} \\
&= (j\omega)^2 V
\end{aligned} \tag{2.35}$$

Thus the frequency domain version of (2.18) is

$$\begin{aligned}
\frac{\partial^2 V}{\partial x^2} &= RGV + (RC + LG)j\omega V + LC(j\omega)^2 V \\
&= [RG + (RC + LG)j\omega + (j\omega L)(j\omega C)]V \\
\therefore \frac{\partial^2 V}{\partial x^2} &= [(R + j\omega L)(G + j\omega C)]V
\end{aligned} \tag{2.36}$$

Combining (2.24) and (2.36) we obtain

$$\frac{\partial^2 V}{\partial x^2} = \gamma^2 V \tag{2.37}$$

Likewise, the transmission line current equation in frequency domain becomes:

$$\frac{\partial^2 I}{\partial x^2} = \gamma^2 I \tag{2.38}$$

where $I = I(\omega) = \mathcal{F}\{i(t)\}$.

2.4.3.1 Solution to Differential Equation in the Frequency Domain

Equations (2.37) and (2.38) represent the transmission line equations in the frequency domain in their general form. In solving the voltage equation, a solution from [12] is assumed of the form:

$$V = Ae^{-\gamma x} + Be^{\gamma x} \quad (2.39)$$

where $Ae^{-\gamma x}$ represents the forward waveform and $Be^{\gamma x}$ represents the reverse waveform.

Assuming the transmission line is perfectly terminated, no reflected wave occurs and therefore the reverse component is zero.

Equation (2.39) then becomes $V = Ae^{-\gamma x}$ where $A = V_0$ is the initial (input) waveform and $e^{-\gamma x}$ shows the exponential decay as the waveform propagates along the transmission line.

The equation then becomes:

$$V = V_0 e^{-\gamma x} \quad (2.40)$$

The transmission line's current differential equation also has the solution:

$$I = I_0 e^{-\gamma x} \quad (2.41)$$

To prove that (2.40) is a solution to (2.37), the former is substituted into the latter.

2.5 Phase Velocity and Propagation Constant

Phase velocity is the speed at which a fixed point on a wave appears to move along the transmission line. The phase velocity can also be seen as the propagation speed of the signal. This is not always the case because other factors such as an absorptive transmission line greatly affects the speed of propagation of the signal along the line.

The phase velocity, V_p , is defined:

$$V_p = \frac{1}{\sqrt{LC}} \quad (2.42)$$

Comparing (2.29) with (2.42) yields

$$V_p = \frac{\omega}{\beta} \quad (2.43)$$

Equation (2.43) shows that the phase velocity depends on frequency since $\omega = 2\pi f$. However, [2], assumes that all parameters are independent of frequency.

The propagation constant is assumed to be the solution of the differential equation of the transmission line.

2.6 Pulses on a Transmission Line

A pulse on a transmission line can be conceptualized as charging and discharging the line through a resistive load in a relatively short duration of time. An electric signal is considered a pulse its width is much more smaller than the physical length of the transmission line.

To study pulses on transmission lines, basically a step function is of prime importance because any pulse can be taught of and analyzed in terms of two step functions, one

positive and one negative. The delay in time between two step functions equals the pulse width as shown in Figure 2.7.

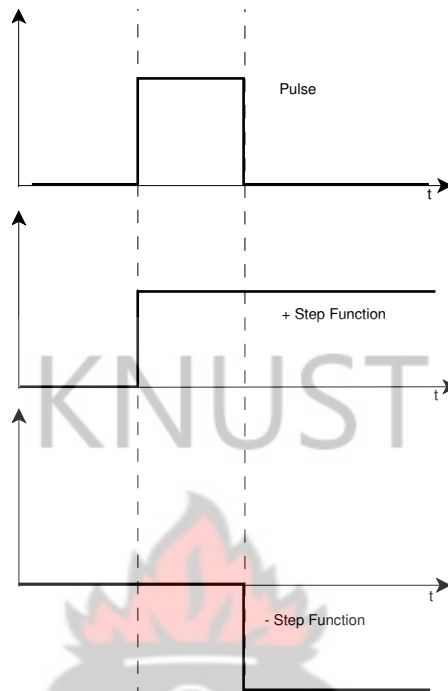


Figure 2.7: Pulse obtained from a superposition of two step functions [2]

The pulse shape at any point on the transmission line is affected by reflections in the transmission line caused by improper terminations at the source and the load. The pulse reflection of a given ideal transmission line can be obtained by the proper superposition of the step function waveforms.

2.6.1 Step and Pulse Response of a Transmission Line

The step function response of a transmission line helps the study of the response of pulses. Figures 2.8 and 2.9 show the response of various circuit models on pulses as stated in [2]. These give an idea of pulse shapes as we pass them through various types of lines.

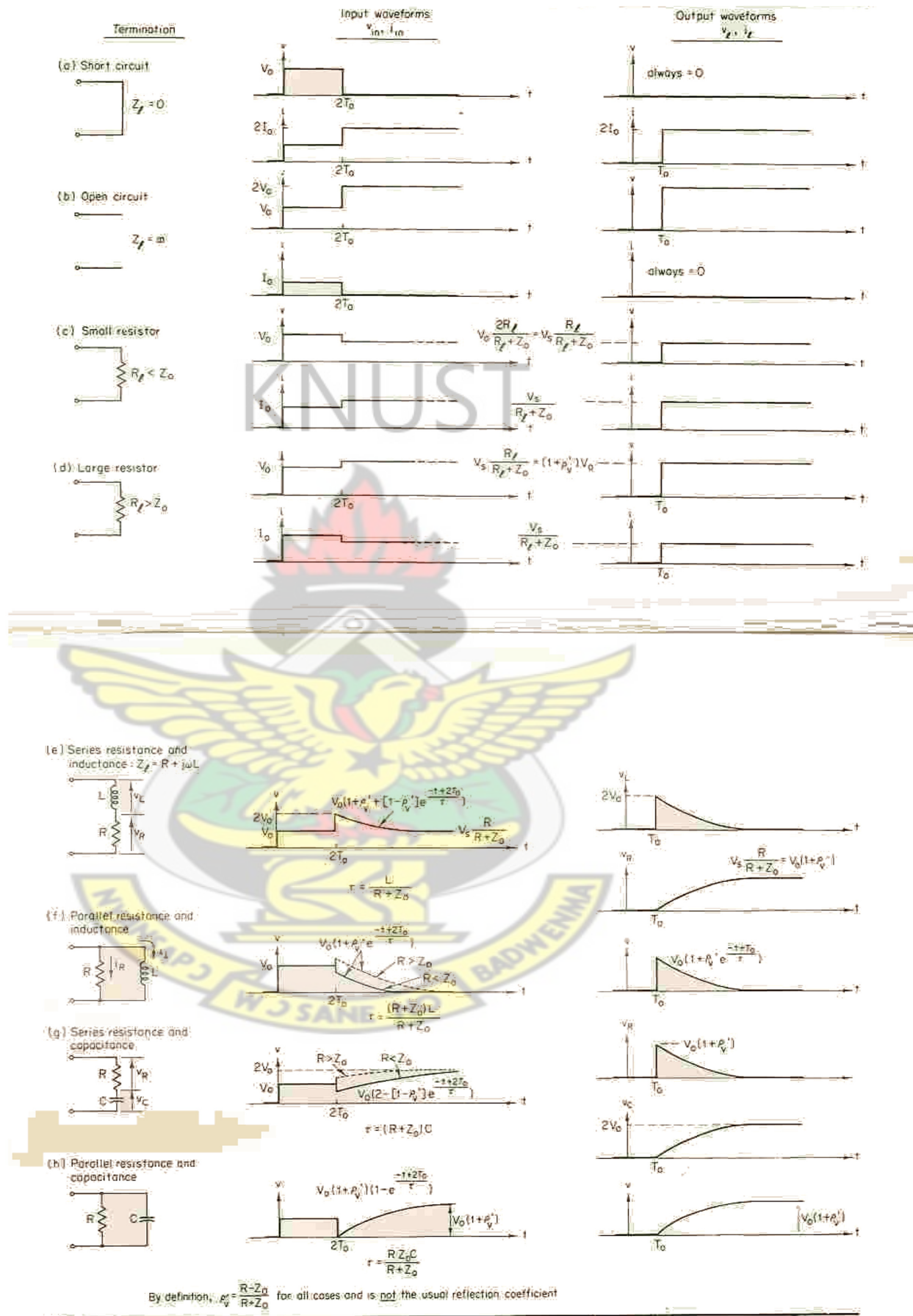


Figure 2.8: Step function response of transmission lines with various terminations [2]

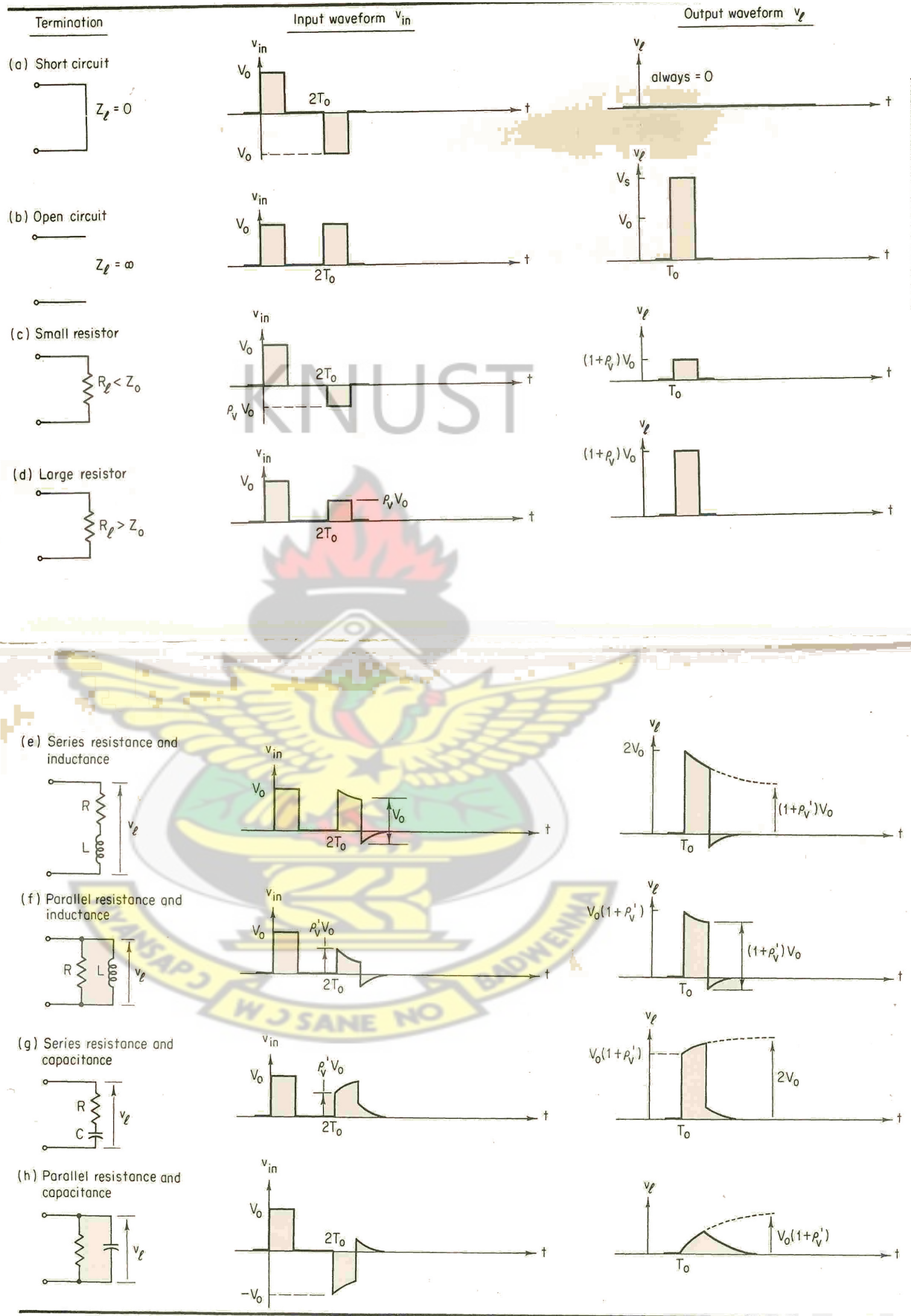


Figure 2.9: Pulse function response of transmission lines with various terminations[2]

The pulse propagation through the transmission line represented in figure 2.9 shows how the line responds to pulses in various lumped models of the line. This thesis focuses mainly on the responses in figure 2.9 (e) and (h) which show the responses of series connected RL and parallel RC circuits.

2.7 Multi-access Transmission Line

A multi-access wireline or transmission line is analogous to the bus network topology where a common transmission line is accessed at different points along the line. A multi-access transmission line is shown in Figure 2.10. The nodes $N_1, N_2, \dots, N_k, N_{k+1}, \dots, N_K$ represents the multiple access nodes. The internode distances are represented by $d_1, d_2, \dots, d_k, d_{k+1}, \dots, d_{K-1}$, where d_{K-1} represents the distance between the N_{K-1} and the N_K node. d_0 represents the distance between the source and the first node; d_K represents the distance between the K th node and the terminating node.

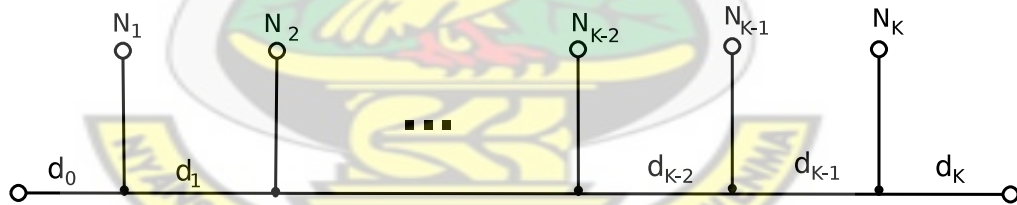


Figure 2.10: Conceptual view of multi-access transmission line

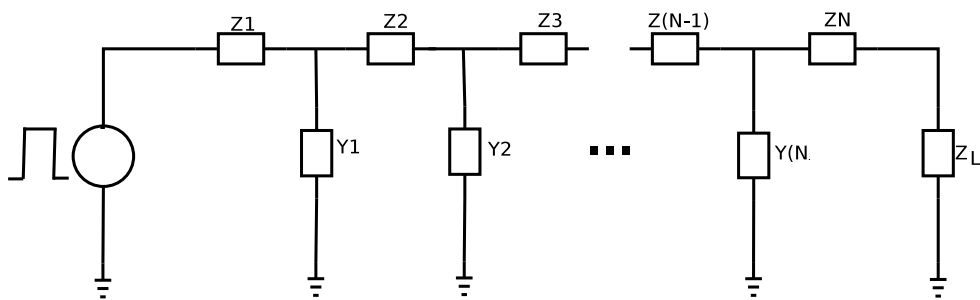


Figure 2.11: Circuit diagram of multi-access transmission line

Figure 2.10 can also be represented by Figure 2.11 which shows the circuit diagram of the conceptual diagram. The impedances of the main sections of the transmission line represented as $Z_1, Z_2, \dots, Z_n, Z_{n+1}, \dots, Z_N$. The input pulse voltage is represented by V_{in} and the matched load impedance is Z_L . The various node impedances are also represented by $Y_1, Y_2, \dots, Y_n, Y_{n+1}, \dots, Y_{(N-1)}$.

2.7.1 Node model of multi-access transmission lines

Each node in the model is considered as a generalized RLC load [6] as shown in Figure 2.12.

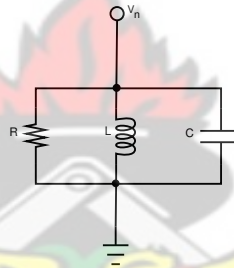


Figure 2.12: Model of transmission line node

This reduces the amount of computation to that of a single branch node. If the branch is considered as a transmission line as the main line the computation becomes much more complex. The reduction of the branch node to a simple parallel RLC circuit means that it only affects the current and not the voltage.

The total susceptance (measures in siemens) of the line then becomes:

$$Y = \frac{1}{R} + j\left(\omega C - \frac{1}{\omega L}\right) \quad (2.44)$$

This then makes the current, I_{bn} dissipated at the branch node

$$I_{bn} = V_n Y \quad (2.45)$$

$$= V_n \left[\frac{1}{R} + j(\omega C - \frac{1}{\omega L}) \right] \quad (2.46)$$

and the current, I_{n+1} , which moves to the rest of the transmission line is

$$I_{n+1} = I_n - I_{bn} \quad (2.47)$$

2.8 Reflection on Transmission Lines

There are reflections in the transmission line if the source impedance or the terminating impedance do not match the characteristics impedance of the transmission line.

Considering a basic loaded transmission line shown in Figure 2.13, if the source impedance, Z_s and the load Z_L are not equal to the characteristics impedance, Z_0 , there will be both source reflection and node reflection.

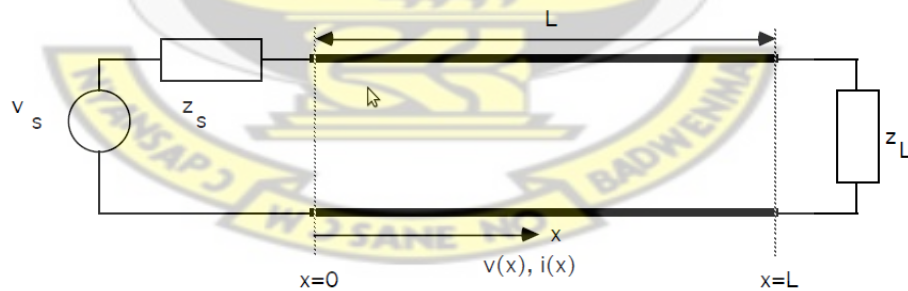


Figure 2.13: Basic Loaded transmission line with length L , input voltage V_s , input impedance Z_s and terminated with load Z_L [3]

From [3], the voltage, $v(x)$ and current $i(x)$ at a point x , on the transmission line is

$$v(x) = \frac{V_s Z_0}{Z_0 + Z_s} e^{-\gamma x} \frac{1 + \Gamma_L e^{-2\gamma(L-x)}}{1 - \Gamma_s \Gamma_L e^{-2\gamma(L-x)}} \quad (2.48)$$

$$i(x) = \frac{V_s}{Z_0 + Z_s} e^{-\gamma x} \frac{1 + \Gamma_L e^{-2\gamma(L-x)}}{1 - \Gamma_s \Gamma_L e^{-2\gamma(L-x)}} \quad (2.49)$$

where Γ_L and Γ_s , the reflection coefficients of the load and source respectively are defined

$$\Gamma_L = \frac{Z_L - Z_0}{Z_L + Z_0} \quad (2.50)$$

$$\Gamma_s = \frac{Z_s - Z_0}{Z_s + Z_0} \quad (2.51)$$

2.8.1 Reflection Coefficient of Nodes on a Transmission Line

For a transmission line with a node impedance of Z_n and the characteristic impedance Z_0 , the reflection coefficient Γ_n at the node of the transmission line is defined by [4] as;

$$\Gamma_n = \frac{(Z_n \parallel Z_0) - Z_0}{(Z_n \parallel Z_0) + Z_0} \quad (2.52)$$

where

$$Z_n \parallel Z_0 = \frac{Z_n Z_0}{Z_n + Z_0} \quad (2.53)$$

Chapter 3

Model Description

In this chapter we review the three composite models derived for the work. The models are:

- Voltage Propagation Model (VPM)
- Current Propagation Model (IPM)
- Power Propagation Model (PPM)

3.1 Model Assumptions

To make it easier to model the system, the following assumptions were made:

1. The tapped transmission line is perfectly terminated both at the source and the load ends.
2. Only a single pulse is transmitted over the line over the length of the transmission line at a time
3. Only the isolated transmitted pulse is studied over the line.
4. Reflections above the second order are not considered.

3.2 Model Description

The transmission line is modeled in the frequency domain to reduce the amount of computation. Figure 3.1 shows multi-access transmission line divided into blocks of transfer functions; $TF_1, TF_2, \dots, TF(n-1), TF(n), TF(n+1), \dots, TF(N-1), TF(N)$.

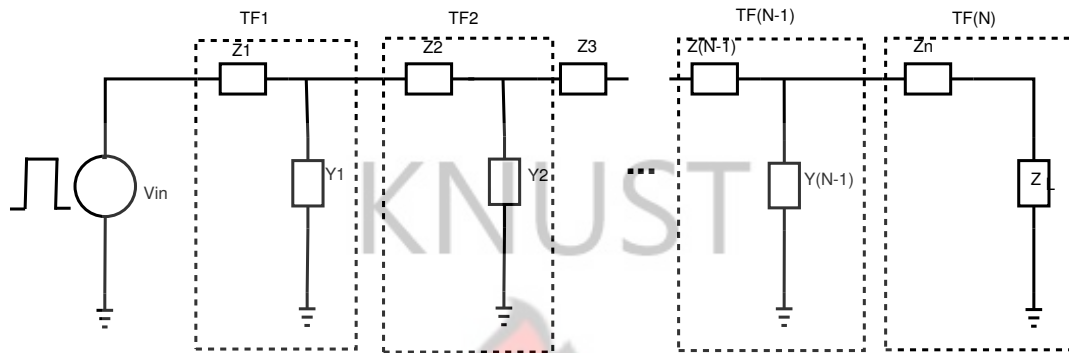


Figure 3.1: Multi-access transmission line divided into blocks

The output of each transfer function shows the output of each segment. The concept of the model is to determine the pulse characteristics at each point on the tapped transmission line as a pulse propagates over the line.

It is difficult to determine the shape of the propagated pulse at every point of the transmission line so the model was discretized for easier calculations.

Since the transmission line is perfectly terminated at both the source and the load ends, a pulse on the line is not affected by reflection at both ends of the line.

In the model, the transmission line is broken into a series of cascaded blocks as shown in Figure 3.2. Each section of the line is made up of a series impedance and a shunt impedance of the transmission line.

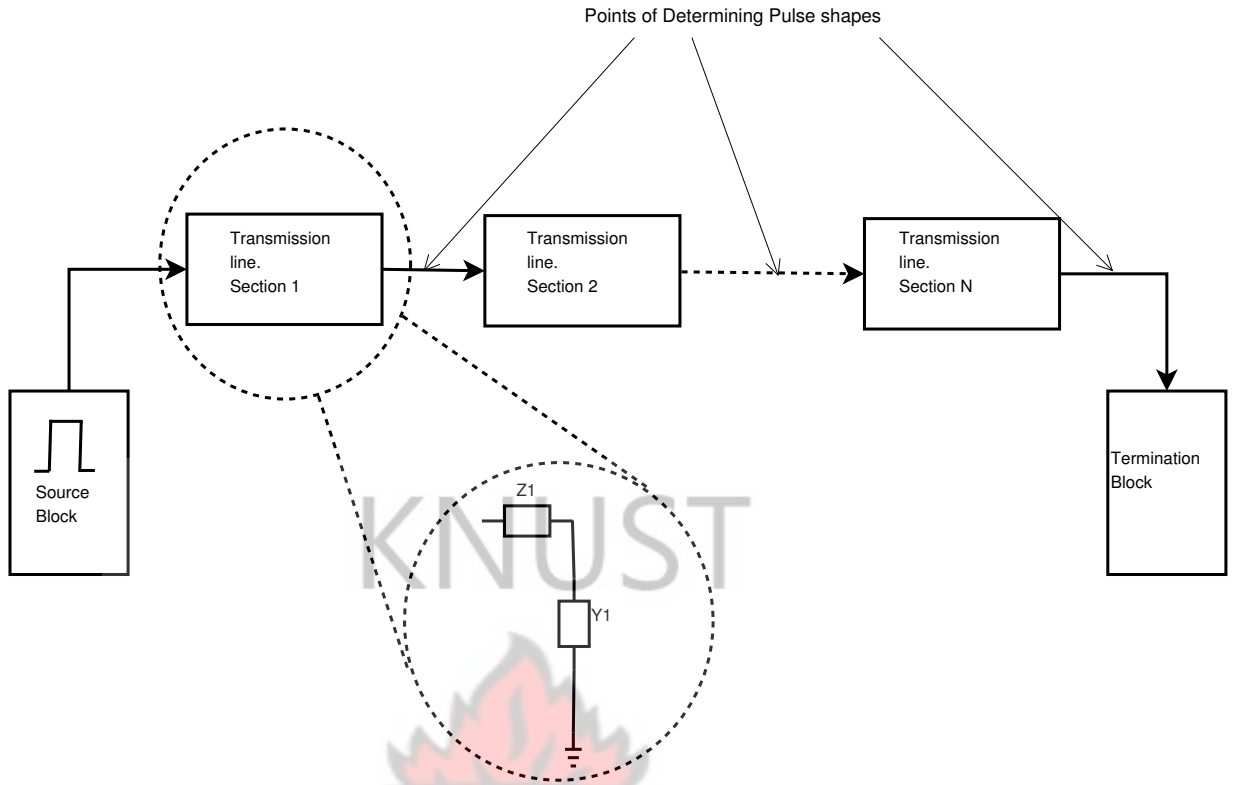


Figure 3.2: Block diagram of multi-access transmission line modeled.

3.2.1 Voltage Propagation Model (VPM)

In the VPM, the voltage pulse shape is observed as it propagate through the transmission line. From Subsection 2.7.1, the branch nodes reflect part of the pulse as it propagates along the length of the transmission line and is therefore simplified as a parallel transmission line as shown in Figure 2.3.

The VPM model is shown in Figure 3.3 where:

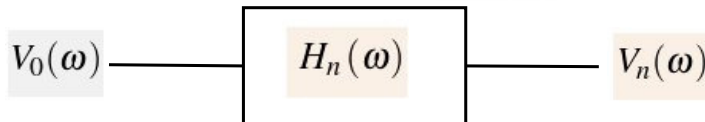


Figure 3.3: Voltage Propagation Model (VPM) in the frequency domain

- $V_0(\omega)$ is the input pulse in the frequency domain.
- $H_n(\omega)$ is the transfer function of the distributed transmission line parameter at position n .

- $V_n(\omega)$ is the output pulse shape of interest in the frequency domain.

3.2.1.1 VPM Equation

From the VPM description,

$$V_n(\omega) = H_n(\omega)V_0(\omega) \quad (3.1)$$

and from (2.40), (3.1) is rewritten simply as equation (3.2) if there are no nodes on the line.

$$V_n = V_0 e^{-\gamma x} \quad (3.2)$$

where:

$$V_n(\omega) = V_n$$

$$V_0(\omega) = V_0$$

$$H_0(\omega) = e^{-\gamma x}$$

Once nodes are present, there is bound to be reflection and the transfer function is no longer as simple as shown in equation (3.2).

Let

V_m be the voltage at the m th position

V_{Nn} be the nodal voltage at the n th node, the node before the m th position

D be the internodal distance

r be the reflection coefficient at the nodes

r_s be the reflection coefficient of the source

T be the transmission coefficient at the nodes

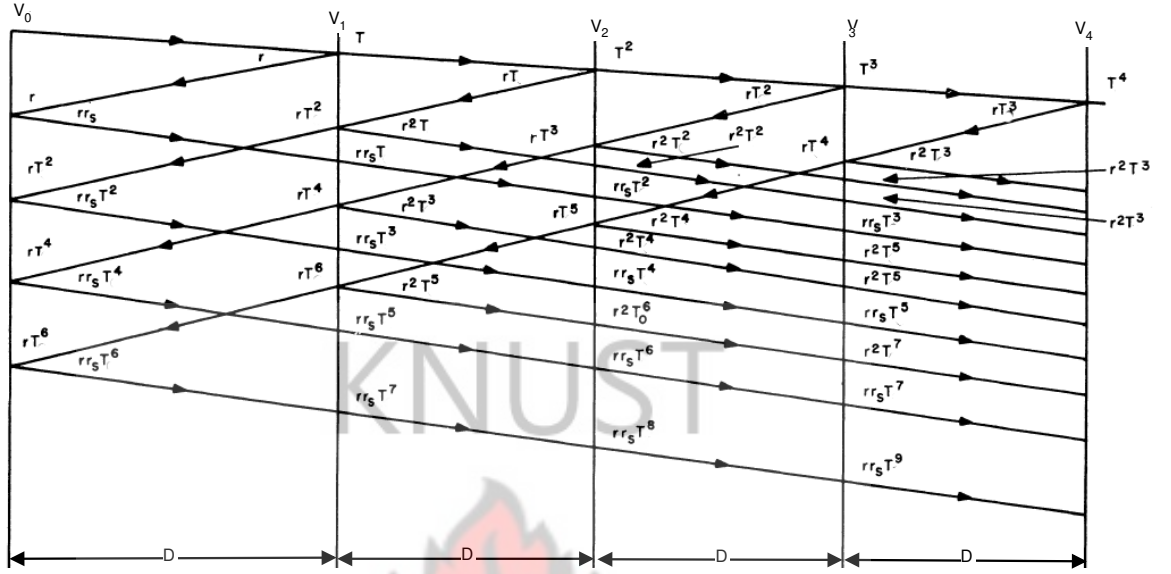


Figure 3.4: Voltage Propagation Model (VPM) tapped transmission line reflection model with 4 nodes [4]

Now from Figure 3.4 we obtain

$$T = 1 + r \quad (3.3)$$

$$\begin{aligned} V_{N1} = & V_0 [Te^{-\gamma D} + rT^2e^{-\gamma 3D} + rr_sTe^{-\gamma 3D} \\ & + rT^4e^{-\gamma 5D} + rr_sT^3e^{-\gamma 5D} + rT^6e^{-\gamma 7D} \\ & + rr_sT^5e^{-\gamma 7D} + rr_sT^7e^{-\gamma 9D}] \end{aligned} \quad (3.4)$$

$$\begin{aligned} V_{N2} = & V_0 [T^2e^{-\gamma 2D} + rT^3e^{-\gamma 4D} + r^2T^2e^{-\gamma 4D} \\ & + rr_sT^2e^{-\gamma 4D} + rT^5e^{-\gamma 6D} + rr_sT^4e^{-\gamma 6D} \\ & + rr_sT^6e^{-\gamma 8D} + rr_sT^8e^{-\gamma 10D}] \end{aligned} \quad (3.5)$$

$$\begin{aligned}
V_{N3} = & V_0[T^3 e^{-\gamma 3D} + rT^4 e^{-\gamma 5D} + r^2 T^3 e^{-\gamma 5D} \\
& + r^2 T^3 e^{-\gamma 5D} + rr_s T^3 e^{-\gamma 5D} + rr_s T^5 e^{-\gamma 7D} \\
& + rr_s T^7 e^{-\gamma 9D} + rr_s T^9 e^{-\gamma 11D}]
\end{aligned} \tag{3.6}$$

$$\begin{aligned}
V_{N4} = & V_0[T^4 e^{-\gamma 4D} + r^2 T^4 e^{-\gamma 6D} + r^2 T^4 e^{-\gamma 6D} \\
& + r^2 T^4 e^{-\gamma 6D} + rr_s T^4 e^{-\gamma 6D} + rr_s T^6 e^{-\gamma 8D} \\
& + rr_s T^8 e^{-\gamma 10D} + rr_s T^{10} e^{-\gamma 12D}]
\end{aligned} \tag{3.7}$$

Now projecting for a transmission line with N nodes, the nodal voltage, V_{Nn} , at the n th node becomes

$$\begin{aligned}
V_{Nn} = & V_0[T^n e^{-\gamma nD} + (n-1)r^2 T^n e^{-\gamma(n+2)D} \\
& + rr_s \sum_{k=0}^{N-1} (T^{n+2k} e^{-\gamma(n+2k+2)D}) \\
& + r \sum_{k=0}^{N-n-1} (T^{n+2k+1} e^{-\gamma(n+2k+2)D})]
\end{aligned} \tag{3.8}$$

Now the voltage pulse at position m on a transmission line with N evenly spaced nodes is

$$V_m = V_{Nn} e^{-\gamma(m-n)} \tag{3.9}$$

From (3.9), the transfer function, $H(\omega)$, becomes

$$H(\omega) = [T^n e^{-\gamma n D} + (n-1)r^2 T^n e^{-\gamma(n+2)D} + rr_s \sum_{k=0}^{N-1} (T^{n+2k} e^{-\gamma(n+2k+2)D}) + r \sum_{k=0}^{N-n-1} (T^{n+2k+1} e^{-\gamma(n+2k+2)D})] e^{-\gamma(m-n)}$$

3.2.2 Current Propagation Model (IPM)

In the IPM model as the name suggests, the current is observed as it propagates through the transmission line. Unlike the VPM model, branch nodes affects the propagation of current as there are branch currents. The IPM model is conceptualized in Figure 3.5

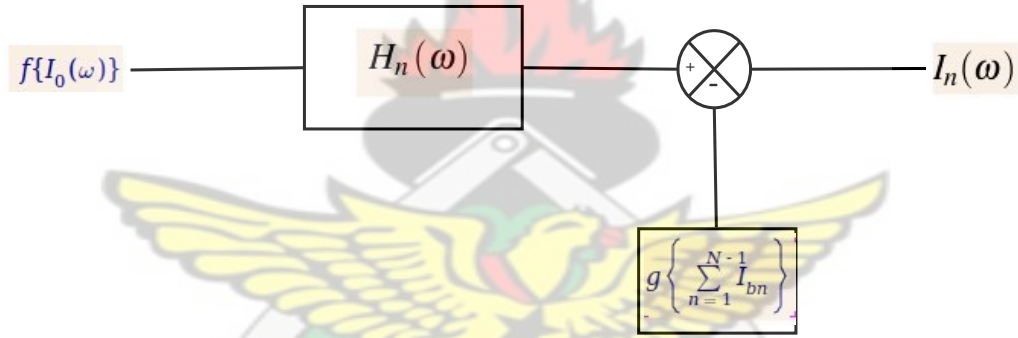


Figure 3.5: Current Propagation Model (IPM) in the frequency domain

where:

$I_0(\omega)$ is the source current in the frequency domain.

$f\{I_0(\omega)\}$ is a function of the input current which changes as the pulse passes each node

$I_n(\omega)$ is the current at position n on the main transmission line.

$I_{bn}(\omega)$ is the branch current at node N

$g\left\{\sum_{n=1}^{N-1} I_{bn}\right\}$ is a sum of $I_{bn}(\omega)$ where each $I_{bn}(\omega)$ is determined by the value of $f\{I_0(\omega)\}$

3.2.2.1 IPM Equation

From the diagram in Figure 3.5, the transmission line is considered in sections.

now,

$$I_{bn} = V_n Y \quad (3.10)$$

and

$$V_n = H_n(\omega) V_0 \quad (3.11)$$

let

$H(\omega)$ be the internode transfer function

and

I_n^+ be the current leaving node N

then

$$I_1 = H(\omega) I_0(\omega) \quad (3.12)$$

$$\begin{aligned} I_1^+ &= H(\omega) I_0(\omega) - I_{b1} \\ &= H(\omega) I_0(\omega) - H_1(\omega) V_0 Y \end{aligned} \quad (3.13)$$

$$\begin{aligned}
I_2^+ &= H(\omega)I_1^+ - I_{b2} \\
&= [H(\omega)I_0(\omega) - H_1(\omega)V_0Y]H(\omega) - H_2(\omega)V_0Y \\
&= H(\omega)^2I_0(\omega) - H(\omega)H_1(\omega)V_0Y - H_2(\omega)V_0Y
\end{aligned} \tag{3.14}$$

$$\begin{aligned}
I_n^+ &= H(\omega)^n I_0(\omega) - V_0Y[H(\omega)^{n-1}H_1(\omega) + H(\omega)^{n-2}H_2(\omega) + \dots \\
&\quad + H(\omega)^2H_{n-2}(\omega) + H(\omega)H_{n-1}(\omega)]
\end{aligned} \tag{3.15}$$

and from equation 2.41,

$$H_n(\omega) = e^{-\gamma n} \tag{3.16}$$

and I_0^1 becomes:

$$I_0 \approx \frac{V_0}{2Z_0} \tag{3.17}$$

also let

D be the average internode distance

and

N is the N th node before the position of interest

¹since the transmission line is terminated with a balanced load, Z_0 , the total impedance on the transmission line circuit becomes $2Z_0$ which is a parallel connection of the characteristic impedance and the terminal impedance

$$H(\omega) = e^{-\gamma D} \quad (3.18)$$

Now from Equation 3.15,

$$\begin{aligned} I_N^+ &= \frac{V_0}{2Z_0} e^{-\gamma DN} - V_0 Y [e^{-\gamma D(N-1)} e^{-D\gamma} + e^{-\gamma D(N-1)} e^{-2D\gamma} + \dots \\ &\quad + e^{-2\gamma D} e^{-\gamma D(N-2)} + e^{-\gamma D} e^{-\gamma D(N-1)}] \\ I_N^+ &= V_0 \left(\frac{e^{-\gamma DN}}{2Z_0} - Y \sum_{k=1}^N e^{-\gamma D(N-k)} e^{-kD\gamma} \right) \end{aligned} \quad (3.19)$$

and

$$\begin{aligned} I_n &= I_N^+ e^{-\gamma(n-N \times D)} \\ \therefore I_n &= V_0 e^{-\gamma(n-N \times D)} \left(\frac{e^{-\gamma DN}}{2Z_0} - Y \sum_{k=1}^N e^{-\gamma D(N-k)} e^{-kD\gamma} \right) \end{aligned} \quad (3.20)$$

3.2.3 Power Propagation Model (PPM)

In the PPM model, the signal power of the input waveform is examined as it propagates along the transmission line. In this model there are three ways in which the power of the propagated signal can be determined;

- Time domain multiple of the voltage, V_n and the current, I_n at the point of interest
- The magnitude square of the of the voltage, $|V_n|^2$ per unit impedance
- The magnitude square of the of the current, $|I_n|^2$ multiplied by the impedance of the line from source to the point of interest

Since the PPM model does not show the pulse shape, it was not used in the simulation though it is worth mentioning

Chapter 4

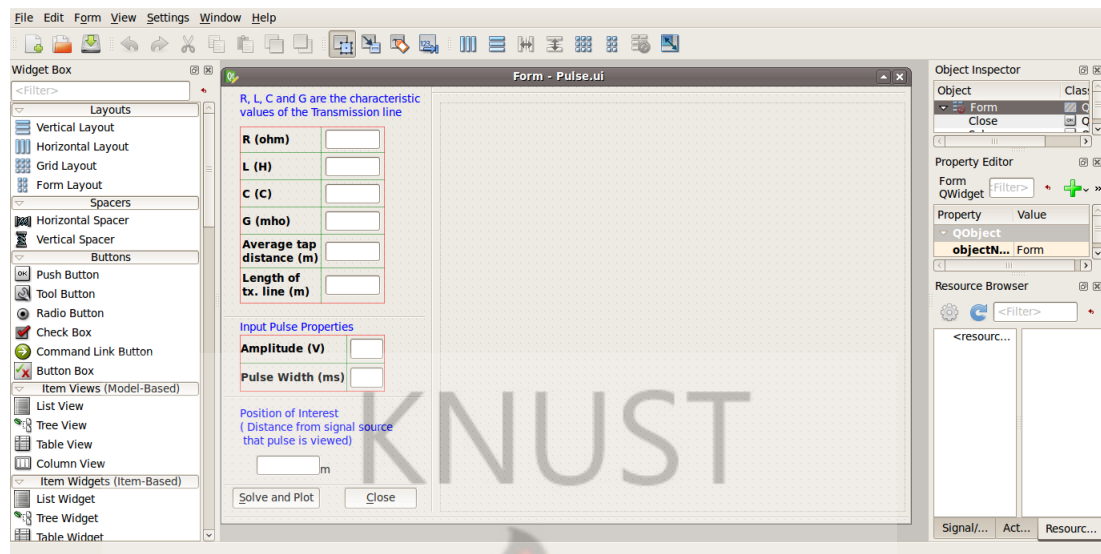
Model Simulation and Results

4.1 Simulation tools

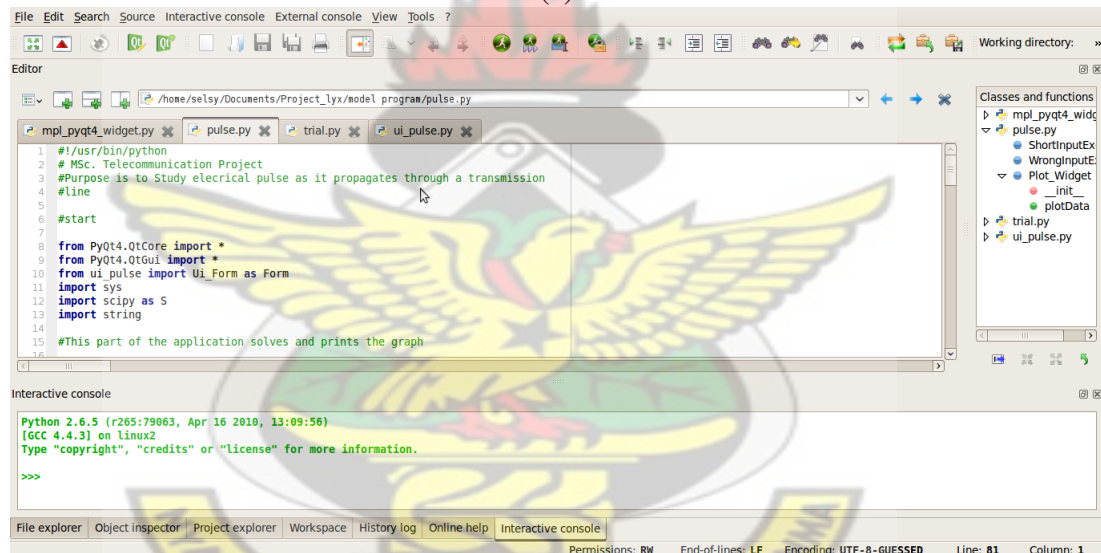
The mathematical models presented in Chapter 3 were simulated using a set of free open source tools. I chose these tools because they are free, well tested and the university has no copy of expensive propriety software such as MatlabTM. These tools are:

- Python with modules Scipy, Numpy, Matplotlib, PyQt : Python is a powerful array manipulation programming language widely used in the open source community. It comes pre-installed on any Linux system and new MacOS systems. Windows versions can be found on the python website. Scipy and Numpy are python modules that allow you to do scientific computing with ease. Matplotlib is also a python module that allows you to plot scientific graphs using command similar to that of MatlabTM. PyQt allows you to call Qt functions and methods from python.
- Qt Designer : Qt Designer is a tool for designing and building graphical user interfaces (GUIs) from Qt components.

- Spyder : Python IDE for scientific computing.



(a)



(b)

Figure 4.1: (a) Qt Designer and (b) Spyder Python IDE

4.2 Voltage Propagation Model Flowchart

The flow chart for the VPM model is obtained in (3.1). The program executes as follows:

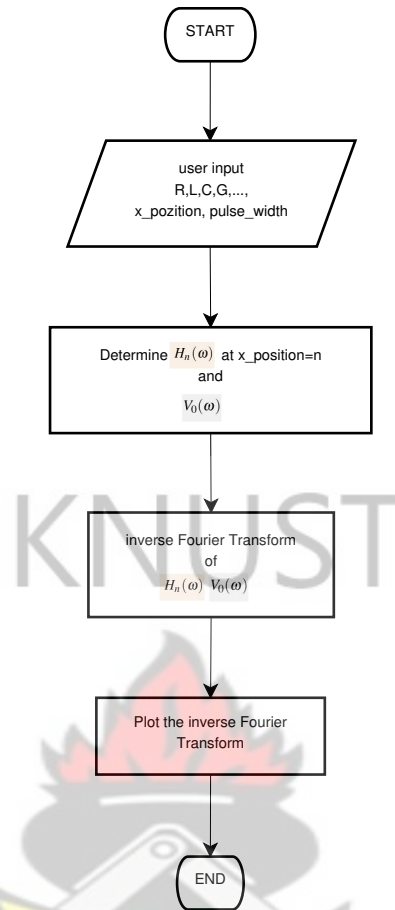


Figure 4.2: Voltage Propagation Model flowchart

1. User inputs the parameters of the transmission line (R , L , C , G), the average tap distance ($x_position$), the width ($pulse_width$) and amplitude of the input pulse, the length of the transmission line and the distance from the signal source where the pulse shaped is determined.
2. The Fourier transform of the input voltage and the tapped transmission line distributed equation are computed
3. The output voltage, $V_n(\omega)$ in the frequency domain is determined by multiplying the parameters in step 2 above.
4. The inverse Fourier transform of the $V_n(\omega)$ is calculated to determine the output voltage, $v_n(t)$
5. Finally $v_n(t)$ is plotted to determine the pulse shape.

4.3 Current Propagation Model Flowchart

The PPM model requires more processing than the voltage propagation model. The PPM model obtained in equation 3.20 is shown in Figure 4.3. The IPM program executes as follows:



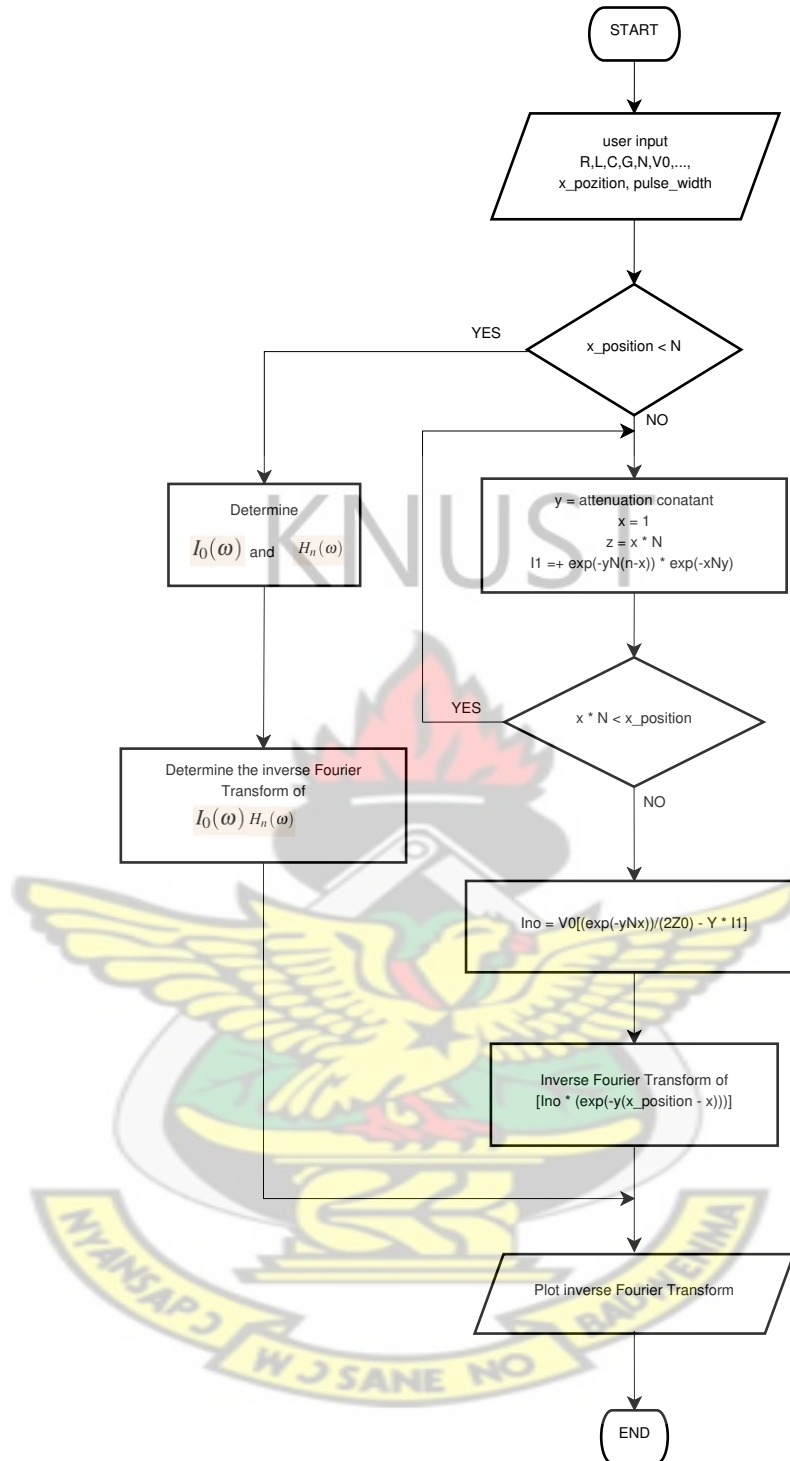


Figure 4.3: Current Propagation Model (IPM) Flowchart

1. The first two steps in section 4.2 above also apply here.
2. If the position of interest ($x_position$) is less than N ; the first tap position, Then

$$I_n = \frac{V_n}{2Z_0} \text{ is then calculated.}$$

3. If $x_{\text{position}} > N$, then I_N which is the current leaving the last node before x_{position} is determined.
4. now $I_n = I_N(\omega)H_k(\omega)$ where k is the difference between the last tap position before x_{position} and x_{position} is determined.
5. $I_n(t)$; the Inverse Fourier Transform of I_n is determined and plotted.

4.4 The GUI of the Model program

The graphical user interface (GUI) of the program is shown in Figure 4.4. The user interface is made up of three parts.

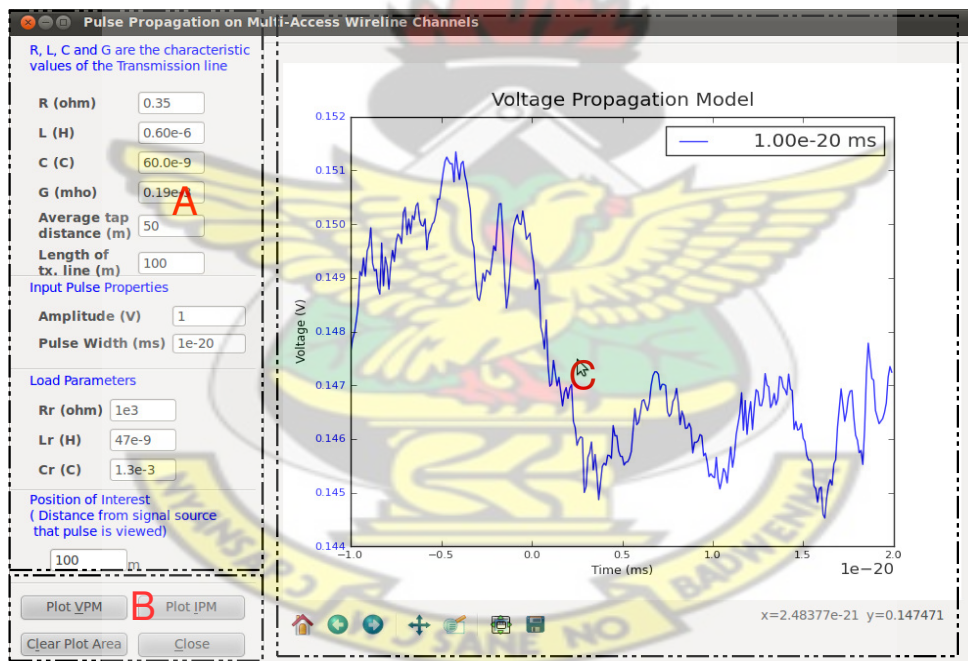


Figure 4.4: Graphical User Interface and entering transmission line parameters and viewing pulse shapes

1. The part for entering the transmission line parameters labeled A.
2. B is used in executing the either the VPM or the IPM models.
3. C is used to display the waveform of the point of interest on the transmission line.

4.5 Simulation Results

Simulation results are shown for various parameter changes for a transmission line with default¹ parameters from [6].

The transmission line parameters are:

$$R = 0.35\Omega/m$$

$$L = 0.60\mu H/m$$

$$C = 60.0pF/m$$

$$G = 0.19mS/m$$

$$Len = 100m$$

The load parameters are:

$$R = 1k\Omega$$

$$L = 1.3mH$$

$$C = 47pF$$

The transmission line is terminated with a resistance of 100Ω and Len is the length of the transmission line.

Figures 4.5 and 4.6 shows the voltage and current pulse shapes for various pulse widths with one load in the middle of the transmission line. Figure 4.7 also shows voltage pulse shapes for various widths when the characteristic impedance of the line is taken to be zero.

¹For simulations where the default parameters change, the values are stated in the graph description.

Figures 4.8 and 4.9 show the IPM pulse shapes with varied number of loads with current pulse width of $1\mu s$ and $1ns$ respectively. Figure 4.8 also shows the magnified form of the graph.

Voltage and Current pulse shapes at various positions on a $100m$ long transmission line are shown in Figures 4.10 and 4.11 respectively with 4.12 and 4.13 showing graphs for varied characteristic inductance and capacitance.

VPM and IPM model pulse shapes are shown in Figures 4.14 and 4.15 with varied characteristic resistance, R and conductance, G .



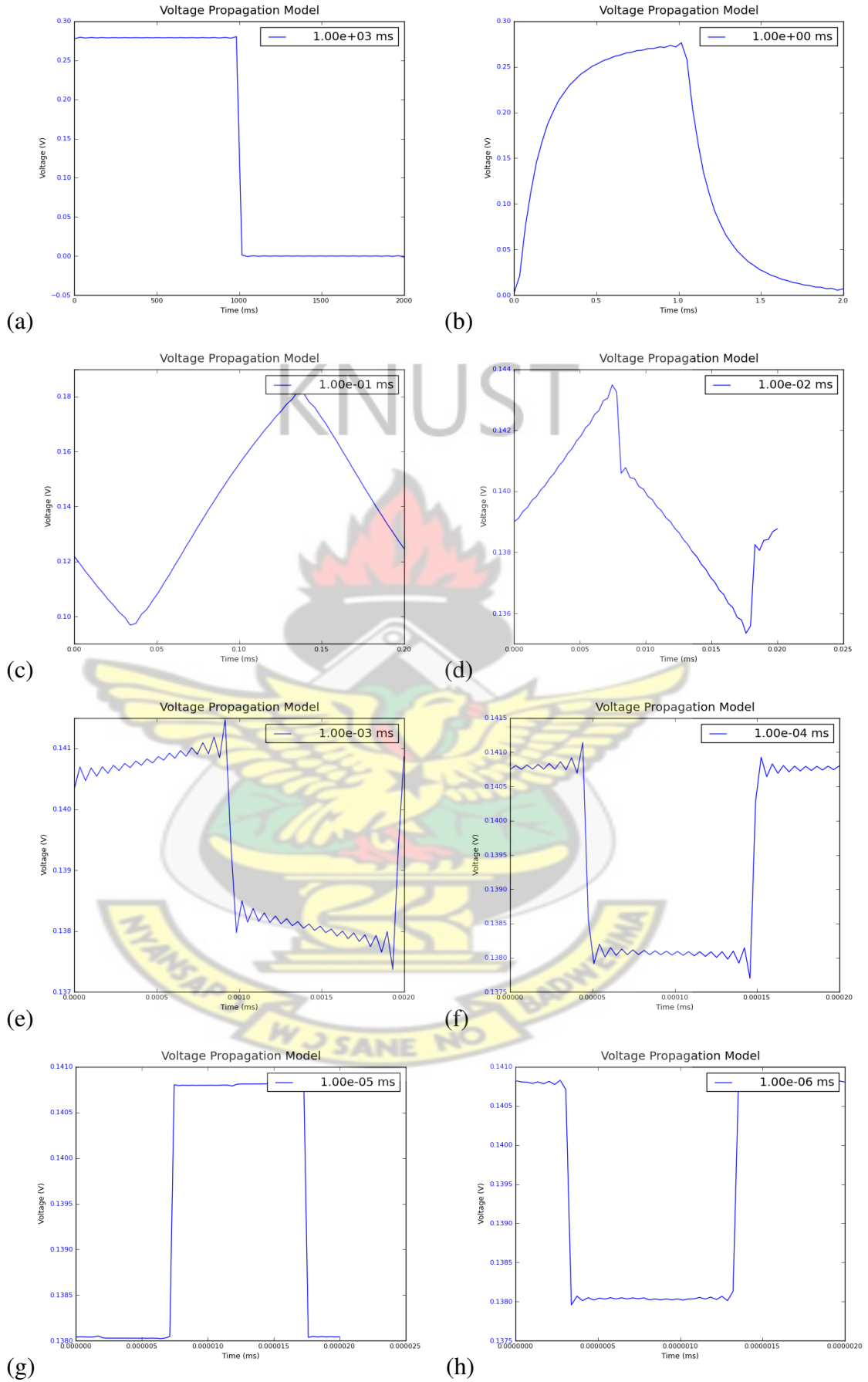


Figure 4.5: Voltage pulse shape at 100m on a 100m transmission line with 1 load in the middle of the transmission line with variable input current pulse width of (a) 1s (b) 1ms (c) $100\mu\text{s}$ (d) $10\mu\text{s}$ (e) $1\mu\text{s}$ (f) 100ns (g) 10ns (h) 1ns

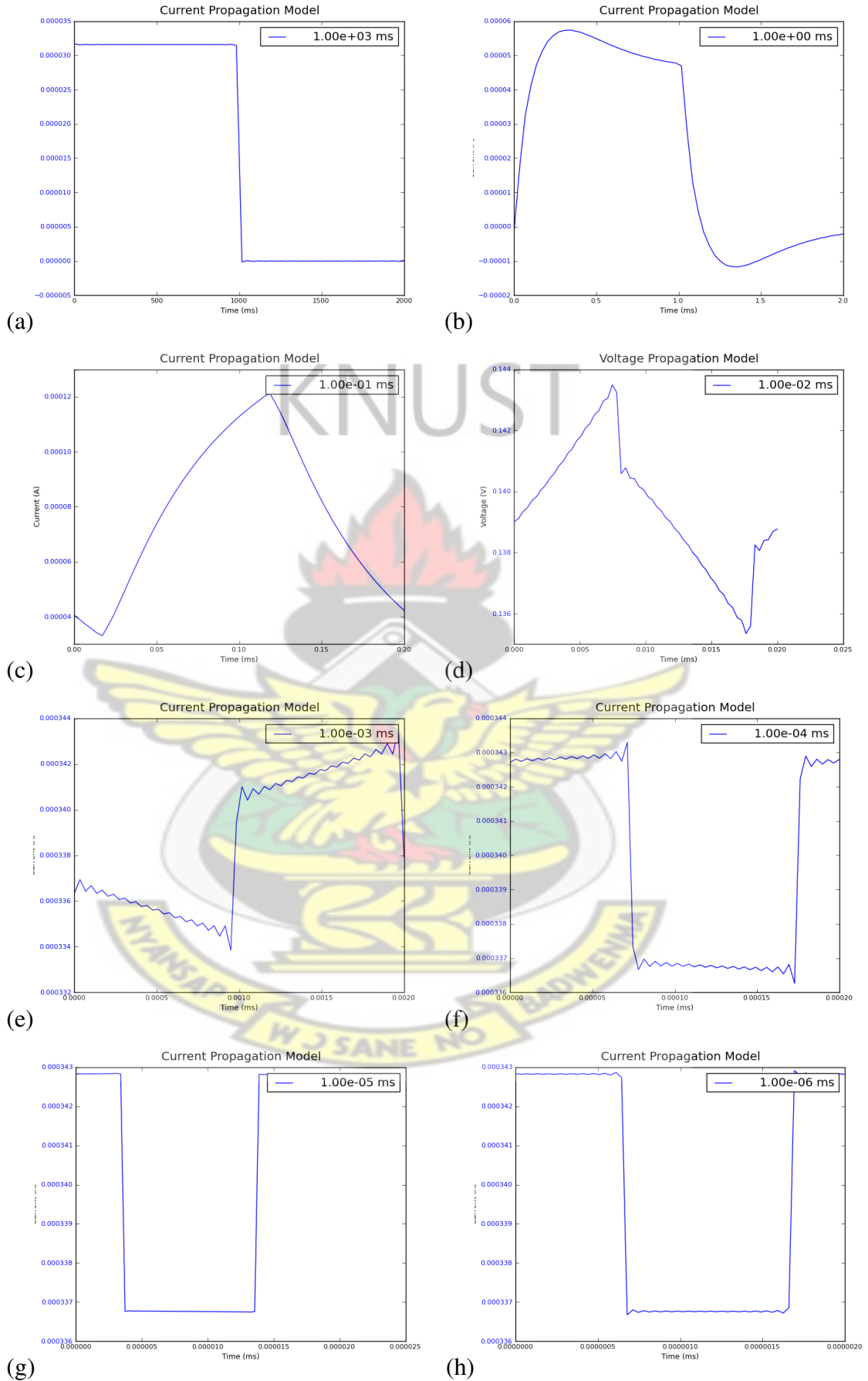


Figure 4.6: Current pulse shape at 100m on a 100m transmission line with 1 load in the middle of the transmission line with variable input voltage pulse width of (a) 1s (b) 1ms (c) 100 μ s (d) 10 μ s (e) 1 μ s (f) 100ns (g) 10ns (h) 1ns

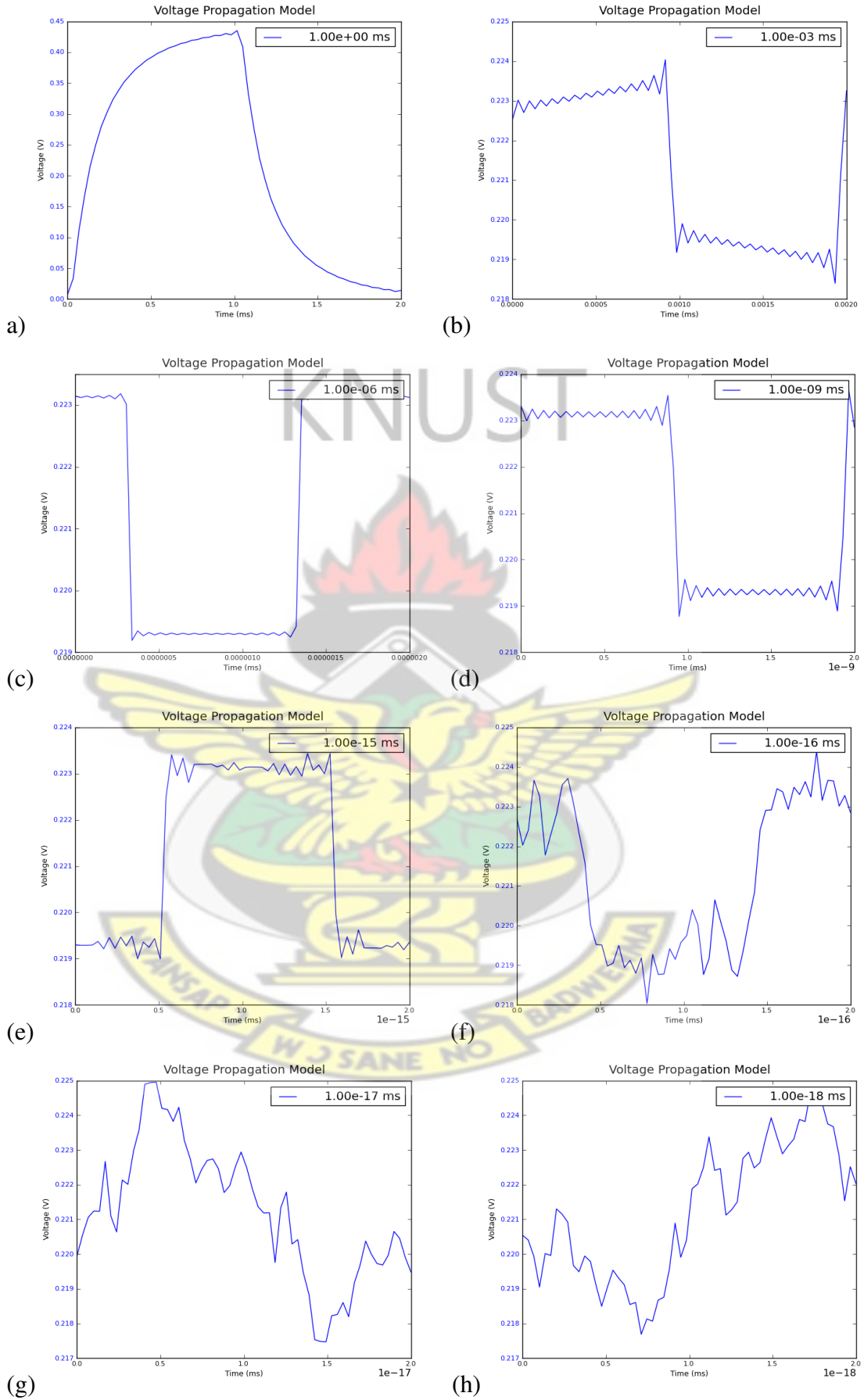


Figure 4.7: Voltage pulse shape at 100m on a 100m transmission line with no nodes on transmission line with variable input voltage pulse width of (a) 1ms (b) $1\mu\text{s}$ (c) 100ps (d) 10fs (e) $1 \times 10^{-15}\text{ms}$ (f) $1 \times 10^{-18}\text{ms}$ (g) $1 \times 10^{-19}\text{ms}$ (h) $1 \times 10^{-20}\text{ms}$

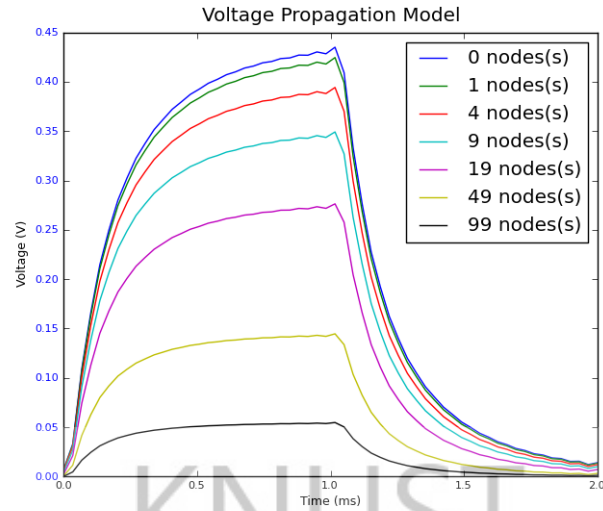


Figure 4.8: Voltage pulse shape at 100m on a transmission line of 100m with input pulse width of 1ms with (a) 0, 1, 4, 19, 49, and 99 nodes on the transmission line

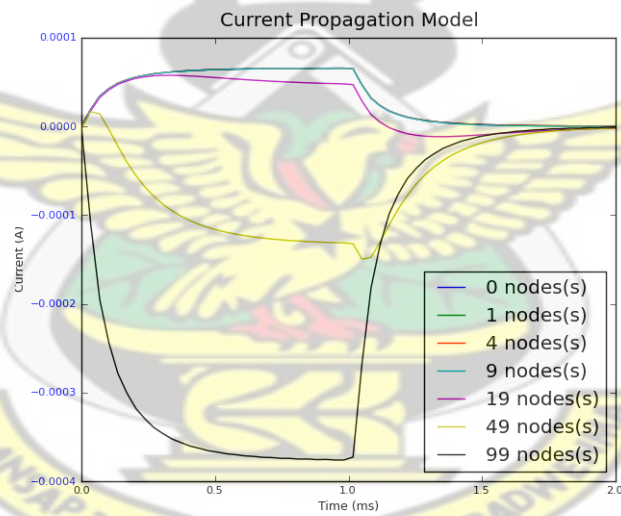


Figure 4.9: Current pulse shape at 100m on a transmission line of 100m with input pulse width of 1ms with (a) 0, 1, 4, 19, 49 and 99 nodes on the transmission line

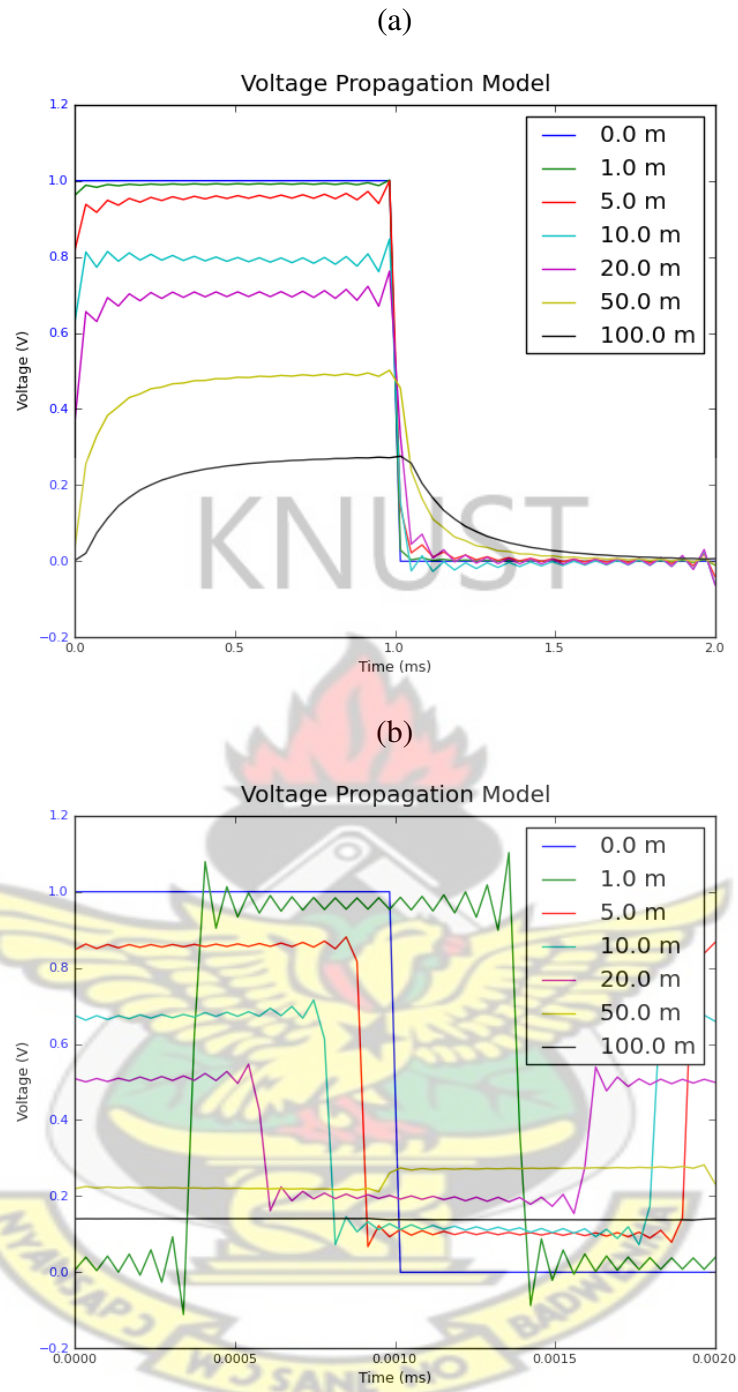


Figure 4.10: Voltage pulse of width (a)1ms and (b)1 μ s on a 100m long transmission line at points on the transmission line

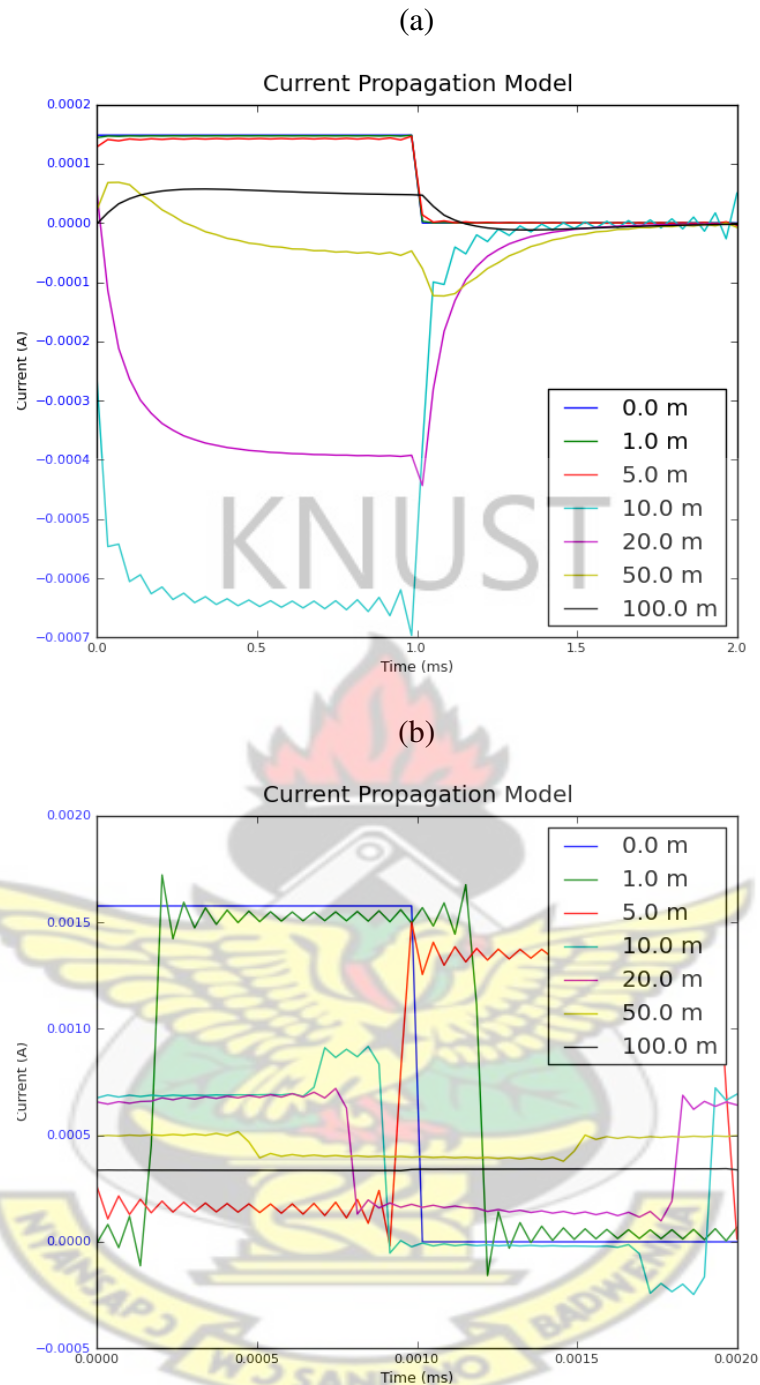


Figure 4.11: Current pulse of width (a) $1\mu\text{s}$ and (b) 1ns on a 100m long at transmission line at points on the transmission line

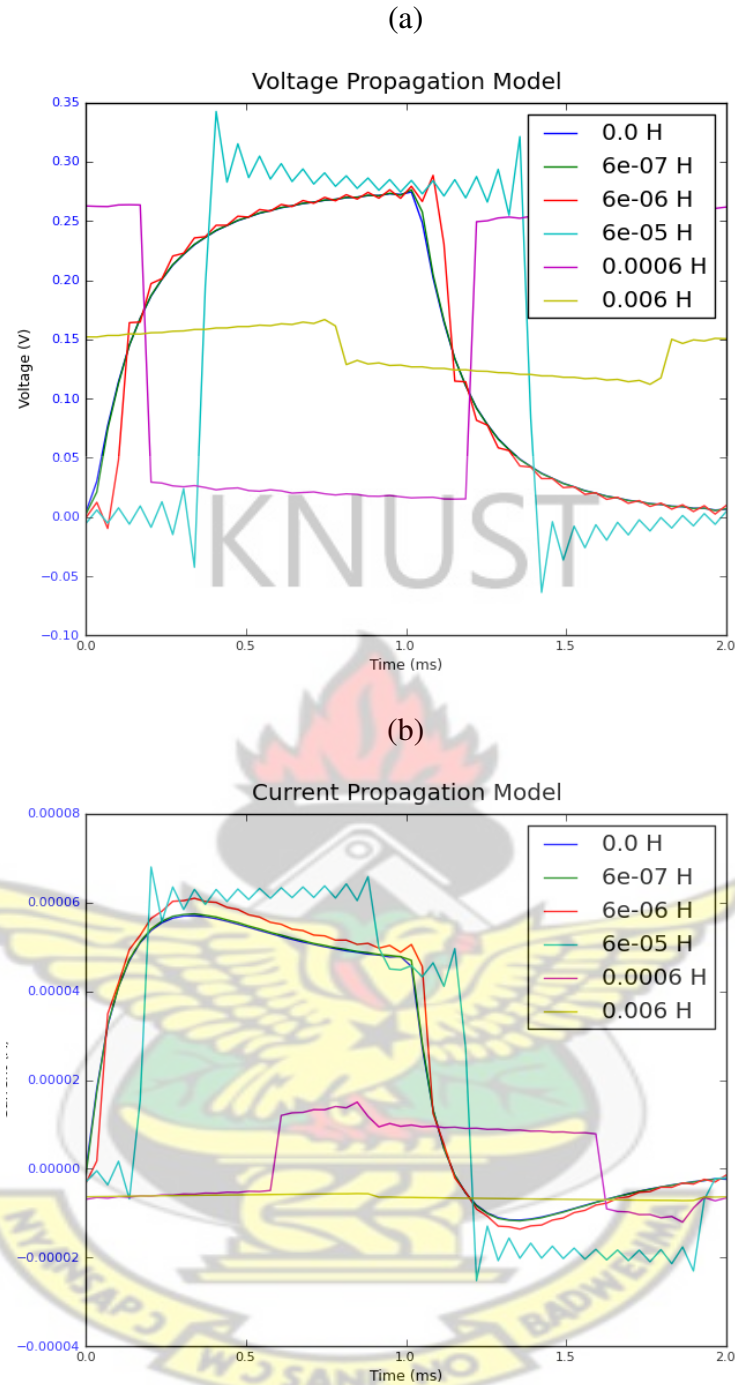


Figure 4.12: Pulse shapes of (a) voltage and (b) current for different characteristic inductance, L values with fixed characteristic capacitance, $C = 60.0\text{pF/m}$

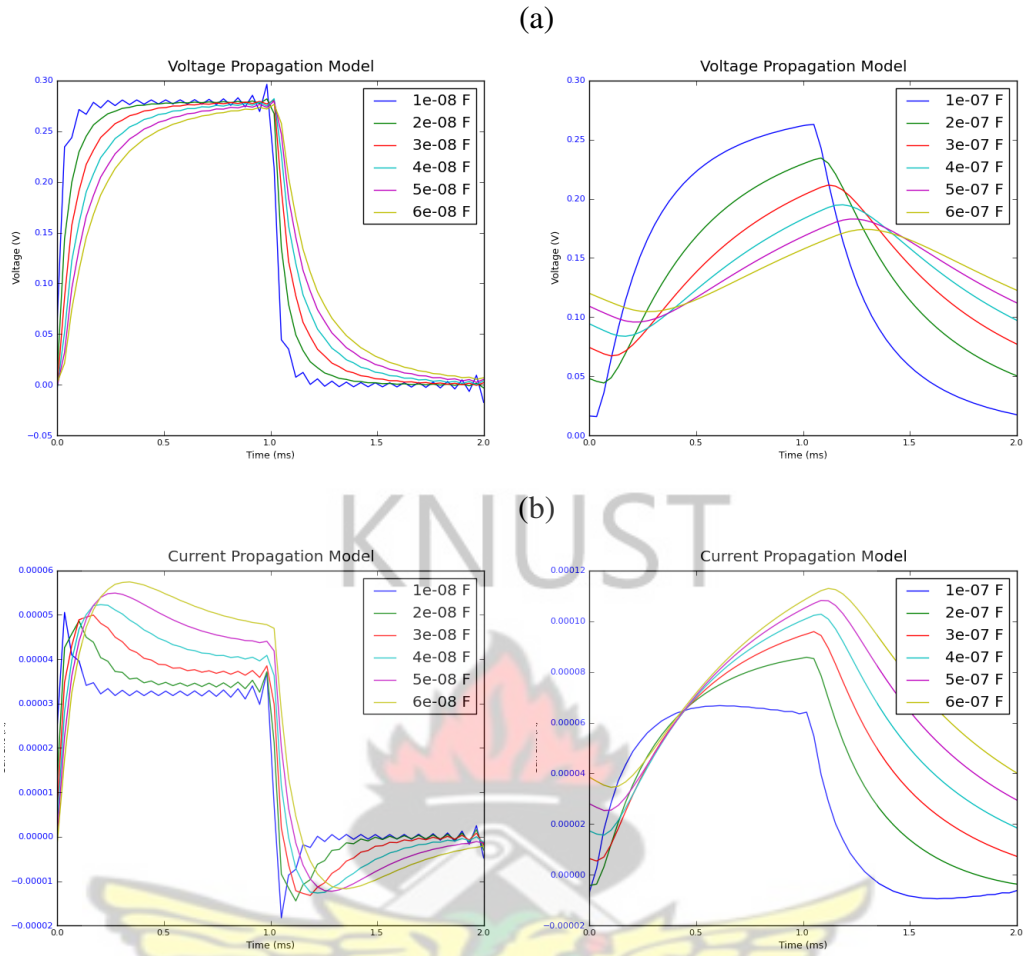


Figure 4.13: Current pulse shapes at 100m for (a) VPM and (b) IPM for variable per unit characteristic capacitances, C with all other parameters remaining constant.

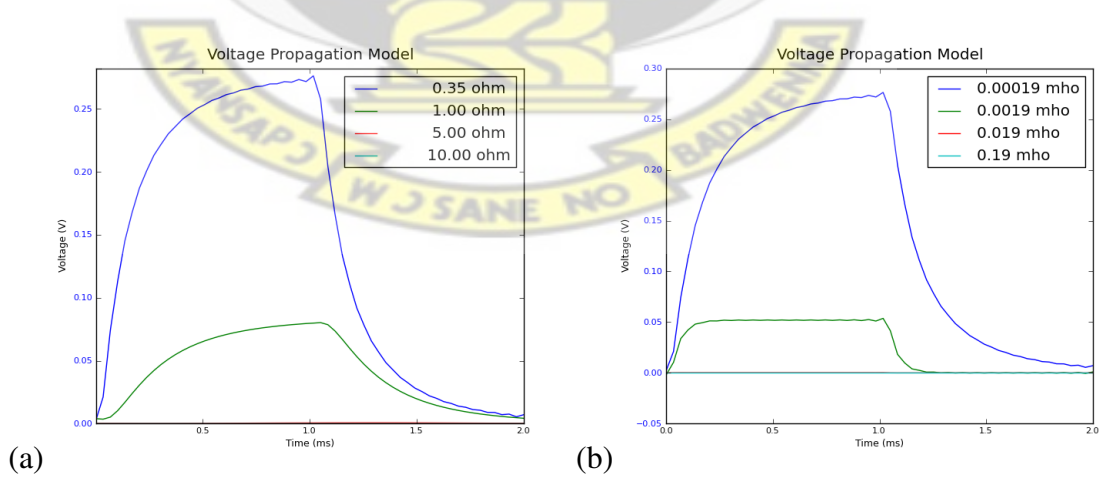


Figure 4.14: Voltage pulse for variable (a) characteristics per unit resistance and (b) characteristics per unit conductance

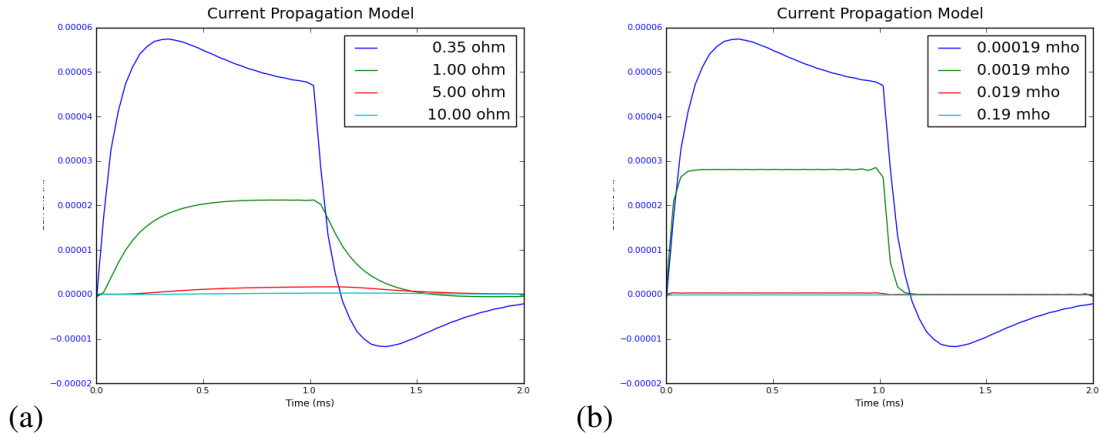


Figure 4.15: Current pulse for variable (a) characteristics per unit resistance, R and (b) characteristics per unit conductance, G

4.6 Results Discussion

4.6.1 Voltage Pulse Model (VPM)

On the effect of pulse width on the transmitted signal, it is seen from Figure 4.5 that pulses are fast attenuated as the frequency increases or as the pulse narrows. The results also show that distortion in the signal increases with decrease in pulse width. This is because narrower pulses tend to attenuate faster and the reactive components of the transmission line affects higher frequencies more.

Figure 4.7 confirms the increase in distortion of narrower pulses when the attenuation (characteristic resistance) is taken out. In the graph, there are much narrower pulses because of the absence of attenuation but no transmission line has the zero attenuation property.

Figure 4.8 shows that the voltage pulse is attenuates more with increase in the number of nodes. This is caused by the increase in the number of reflections as the nodes increase.

Figure 4.10 shows that as the voltage pulse narrows, the pulse tends to spread more. From the Figure, it is seen that the $1\mu\text{s}$ pulse does not spread but the 1ns pulse spreads throughout and does not have any zero value during the duration of the pulse. This occurs because pulses are constituted by waves of different frequencies and therefore tend to propagate at different velocities. Each component arrives at the point of interest at different times and reconstitute into a signal which has been spread. This makes the spreading more noticeable as the signal propagates further away from the source.

Figure 4.12 (a) shows the effect on the voltage pulse as the characteristic inductance changes. The signal spread increases as the inductance increases because of the increasing storage capacity of the inductor. Increasing the capacitance also has a similar effect on the signal but very minimal as shown in Figure 4.13(a).

Increasing the characteristic resistance or conductance attenuates the signal greatly. Figure 4.14 shows the steep reduction in the amplitude when the resistance or conductance is increased. This is because the resistive components dissipate most of the power that is lost.

4.6.2 Current Propagation Model (IPM)

The IPM model generally has the same property as the VPM model. One difference is illustrated in Figure 4.11 where it is shown that the current pulse spreads much more quickly than the voltage pulse as it propagates along the transmission line.

Figures 4.8 and 4.9 show that the amplitude of the current pulse decreases with increase in the number of load nodes. The attenuation also increases with decreasing current pulse width. The current pulse however narrows as the characteristic resistance of the transmission line as shown in Figure 4.15.

Chapter 5

Conclusion and Recommendation

5.1 Conclusion

The thesis focused on voltage and current pulse propagation through a transmission line. The objective was to use the models derived in this work to study the effect of various parameter changes on pulses of different widths. and this objective has therefore been met.

The model results show that longer pulse are better suited for longer distances because they experience less attenuation and distortion. Longer pulses also carry less data so compromise will have to be made between distance and capacity. The model results also show that pulses are greatly distorted when the line capacitance or the inductance or both increase. These results show that to prevent distortion of transmitted signals, the reactive components of the transmission line should be minimized.

5.2 Recommendation

The model obtained should be tested more extensively before being used in real-life applications. The model program is capable of determining more transmission line

characteristics and can therefore be used in other studies. The model is recommended for use in planing the deployment of Power Line Communication (PLC) equipments where sockets parameters are the node parameters of this model.

KNUST



References

- [1] B. C. Wasell, *Transmission Line Design Handbook*. Artech House, 1991.
- [2] R. E. Matick, *Transmission Lines for Digital and Communication Networks (An Introduction to Transmission Lines, High-frequency and High-speed Pulse Characteristics and Applications)*. 3 Park Avenue, 17th Floor, New York, NY, 10016-5997: Institute of Electrical and Electronic Engineers, Inc., an IEEE Press Classic Reissue ed., 1995.
- [3] S. Ardalan and G. Ybarra, "Computer modeling and analysis of plane wave propagation through plasmas," Raleigh, NC27695, 1990.
- [4] F. C. Yao, "Analysis of signal transmission in ultra high speed transistorized digital computers," in *IEEE Transactions on Electronic Computers*, pp. 372–382, Aug. 1963.
- [5] W. D. Block, R. Pintelon, and Y. Rolain, "Identification of lines: From time domain measurements to frequency domain models," in *IEEE Instrumentation and Measurements Technology Conference*, 21-23 May 2002.
- [6] J. K. Oppong, "Loaded highspeed computer data highway: A model to aid designers," *Journal of the Institute of Electronic and Radio Engineers*, vol. 55, pp. 37–41, Jan. 1985.
- [7] J. Anatory, M. M. Kissaka, and N. H. Mvungi, "Channel model for broadband power-line communication," *IEEE Transactions on Power Delivery*, vol. 22, pp. 135–141, Jan. 2007.

- [8] “Principles of transmission lines,” Aug. 2009.
- [9] A. Lehtovuori, “Time-domain model of dispersive transmission line,” Master’s thesis, Helsinki University of Technology, May 2000.
- [10] D. K. Cheng, *Field and Wave Electromagnetics*. USA: Addison-Wesley, 1989.
- [11] H. Meng, S. Chen, Y. L. Guan, C. L. Law, P. L. Sow, E. Gunawan, T. T. Lie, *et al.*, “Modelling of transfer characteristics for the broadband power line communication channel,” *IEEE Transactions on Power Delivery*, vol. 19, pp. 1057–1064, July 2004.
- [12] “http://en.wikibooks.org/wiki/communication_systems/wired_transmission,” Aug. 2009.
- [13] S. K. Mitra, *Digital Signal Processing A Computer-Bssed Approach*. 1221 Avenue of the Americas, NY, 10020: McGraw-Hill Irwin, second ed., 2001.
- [14] S. Salivahanaan, A. Vallavaraj, and C. Gnanapriya, *Digital Signal Processing*. 7 West Patel Nagar, New Delhi 110 008: Tata McGraw-Hill Publishing Company Limited, 2000.

Appendix A

Code Listings

All the source codes used are shown below.

A.1 Main Program Listing

```
1  #!/usr/bin/python
2  # -*- coding: utf-8 -*-
3  #=====
4  # Andrew Selasi Agbemenu
5  # MSc. Telecommunication Project
6  # Pulse Propagation over Multi-Access Wireline Channels
7  #=====
8
9  #=====
10 ## START Program
11 #=====
12
13 from PyQt4.QtCore import *
14 from PyQt4.QtGui import *
15 from ui_pulse import Ui_Form as Form
16 import sys
17 import scipy.signal as S
18 import scipy.fftpack as F
19 import numpy as np
20 import matplotlib.pyplot as plt
21 import string
```

```

22 pi = np.pi
23 a = 30
24
25 #=====
26 # Exception Handling
27 #=====
28
29
30
31 class WrongInputException(Exception):
32     def __init__(self, atleast):
33         Exception.__init__(self)
34         self.atleast = atleast
35
36
37
38 class Plot_Widget(QWidget, Form):
39
40
41     def __init__(self, data2plot = None, parent = None):
42         super(Plot_Widget, self).__init__(parent)
43         self.setupUi(self)
44
45
46         QObject.connect(self.VPM, SIGNAL("clicked()"), self.plotVPM)
47         QObject.connect(self.IPM, SIGNAL("clicked()"), self.plotIPM)
48         QObject.connect(self.Clear, SIGNAL("clicked()"), self.graphClear)
49
50 #=====
51 # #Start graphClear function
52 #=====
53
54     def graphClear(self):
55
56         QMessageBox.warning(self, "Alert", "Matplotlib_Plot_area"
57                               "_will_be_cleared_and_restored_to_default_values")
58
59         self.plotWidget.canvas.ax.cla()
60         self.plotWidget.canvas.format_labels()
61         self.plotWidget.canvas.draw()
62
63         #QMessageBox.warning(self, "Information", "Matplotlib Plot area"
64                               # "cleared")
65

```

```

66  #=====
67  # #End graphClear function
68  #=====
69
70  #=====
71  # #Voltage Propagation Model (VPM) procedure
72  #=====
73
74  def plotVPM(self):
75
76
77      try:
78
79          R = string._float(self.R.text())
80          L = string._float(self.L.text())
81          C = string._float(self.C.text())
82          Rr = string._float(self.Rr.text())
83          Lr = string._float(self.Lr.text())
84          Cr = string._float(self.Cr.text())
85          G = string._float(self.G.text())
86          D = string._float(self.D.text())
87          N = string._float(self.N.text())
88          V_max = string._float(self.V_max.text())
89          Pulse_width = string._float(self.Pulse_width.text())
90          x_position = string._float(self.x_position.text())
91
92          Nt = D/N - 1  # Total number of nodes
93
94
95          if D < N or D < x_position:
96              raise WrongInputException(D)
97
98
99      except WrongInputException, e:
100          QMessageBox.warning(self, "Input_Error", "N,the_average_tap_"
101                                "distance_and_x_postion,the_point_of_interest_cannot_be_"
102                                "greater_than_then_length_of_the_transmission_line,D")
103
104
105      except:
106          QMessageBox.warning(self, "Input_Error", "The_Input_should_be"
107                                "_a_positive_rational_number")
108
109

```



```

110
111
112
113     else:
114         T = Pulse_width
115         Pulse = np.concatenate((S.boxcar(a), np.zeros(a))) * V_max
116         Vin = F.fft(Pulse)
117         c = len(Vin)
118         #t = np.arange(-c/3, 2 * c/3) * (T/a)
119         t = np.arange(0, c) * (2*T/(c-1))
120         freq = 2000 * pi * c / T
121         w = F.fftfreq(c) * freq
122         atten = np.sqrt((R + (w * L) * 1j) * (G + (w * C) * 1j))
123         #Vn = Vin * np.exp(-(x_position * atten))
124         #V = F.ifft(Vn)
125         wl = 2000 * pi / T
126         #wl = w
127         #Z0 = np.abs(np.sqrt((R + 1j * wl * L) / (G + 1j * wl * C))) * D
128         Z0 = R * D
129         print "Z0=_"
130         print Z0
131         g = D/N - 1
132         rs = 1      #reflection coefficient at the source
133
134         if x_position > N and N != 0:
135
136             x = 0
137             q = 1
138             i = 0
139
140             ZL = np.abs(Rr + 1j * (wl * Lr - 1/(wl * Cr))) #impedance of tap branch
141
142             r = ((Z0 * Rr)/(Rr + Z0) - Z0)/((Z0 * Rr)/(Rr + Z0) + Z0) # reflection coefficient
143             T = 1 + r
144             print "r"
145             print r
146
147             while q * N < x_position:
148                 q += 1
149
150                 n = q - 1
151
152                 Vnode1 = (T**(q-1)) * (np.exp(-(atten * (q-1) * N))) + (q-2) * r**2 * T**(q-1) * np.e

```

Node voltage component 1

```

153         Vnode2 = 0 # node voltage component 2 initializatioin
154         Vnode3 = 0 # node voltage component 3 initialization
155
156
157
158
159         while x <= Nt - 1:
160
161             Vnode2 += (T**(q-1 + 2 * x)) * (np.exp(-( atten * N * (q + 1 + 2 * x))))
162             x += 1
163
164         Vnode2 = r * rs * Vnode2
165
166         while i <= Nt - q:
167
168             Vnode3 = (T**(q + 2 * i)) * (np.exp(-( atten * N * (q + 1 + 2 * i))))
169             i += 1
170         Vnode3 = r * Vnode3
171         #print "vnode 1",Vnode1," vnode 2 ", Vnode2,"vnode 3 ", Vnode3
172
173
174         Vn = Vin * (Vnode1 + Vnode2 + Vnode3) * np.exp(-(atten * (x_position - N * (q-1))))
175         #print "Vnode1 + Vnode2 + Vnode3", Vnode1 + Vnode2 + Vnode3
176         #print "Vn", Vn
177
178         #Vn = (Vnode1 * np.exp(-(atten * (x_position - N * (q-1))))) + (Vnode2 * np.exp(-(att
179         V = F.ifft(Vn)
180         #print "V", V
181
182     else:
183
184         Vn = Vin * np.exp(-(x_position * atten))
185         V = F.ifft(Vn)
186
187
188
189         self.plotWidget.canvas.ax.plot(t, np.real(V), label = '%12e_ms'%Pulse_width)
190         #self.plotWidget.canvas.ax.plot(t, V, label = '%d nodes(s)'%g)
191         self.plotWidget.canvas.ax.legend(loc='upper_right')
192         self.plotWidget.canvas.ax.set_xlabel('Time_(ms)', fontsize = 9)
193         self.plotWidget.canvas.ax.set_ylabel('Voltage_(V)', fontsize = 9)
194         # self.plotWidget.canvas.ax.set_ylim(-0.05,0.5)
195         self.plotWidget.canvas.ax.set_title('Voltage_Propagation_Model')
196         self.plotWidget.canvas.draw()

```

[illegible]

```

241
242
243
244
245
246     else :
247
248         T = Pulse_width
249         freq = 2000 * pi * a / T
250         Pulse = np.concatenate((S.boxcar(a), np.zeros(a))) * V_max
251         Vin = F.fft(Pulse)
252         c = len(Vin)
253         #t = np.arange(-c/3, 2 * c/3) * (T/a)
254         t = np.arange(0, c) * (2*T/(c-1))
255         w = F.fftfreq(c) * freq
256         atten = np.sqrt((R + (w * L) * 1j) * (G + (w * C) * 1j))
257         w1 = 2000 * pi / T
258         Y = R**(-1) + 1j * (w1 * Cr - (w1 * Lr)**(-1))
259         Z0 = np.sqrt((R + 1j * w1 * L) / (G + 1j * w1 * C)) * D
260         g = D/N - 1
261
262         if x_position > N and N != 0:
263
264             x = 1
265             q = 1
266             Inode = 0
267
268             while q * N < x_position:
269                 q += 1
270                 print q
271                 while x * N < x_position:
272                     z = x * N
273                     Inode1 = Inode + (np.exp(-(atten * N * (((q-1) * N) - x)))) * (np.exp(-(atten * N * x)))
274                     x = x + 1
275
276                 Inode = Vin * ((np.exp(-(atten * z)))/(2 * Z0) - Y * Inode1)
277
278                 In = F.ifft(Inode * (np.exp(-((x_position - z) * atten))))
279
280         else :
281
282             In = F.ifft(Vin * ((np.exp(-(x_position * atten)))/(2 * Z0)))
283
284         self.plotWidget.canvas.ax.plot(t, In, label = '%12.2e_ms'%Pulse_width)

```

```

285         self.plotWidget.canvas.ax.plot(t, In, label = '%d nodes(s)'%g)
286         self.plotWidget.canvas.ax.set_xlabel('Time_(ms)', fontsize = 9)
287         self.plotWidget.canvas.ax.set_ylabel('Current_(A)', fontsize = 9)
288         self.plotWidget.canvas.ax.set_title('Current_Propagation_Model')
289     #         self.plotWidget.canvas.ax.set_ylim(-0.0005,0.003)
290         self.plotWidget.canvas.ax.legend(loc='lower_right')
291         self.plotWidget.canvas.draw()
292
293     #=====
294     # # END IPM function
295     #=====
296
297     #=====
298     # Call GUI into a loop
299     #=====
300
301     if __name__ == "__main__":
302         app = QApplication(sys.argv)
303         plot = Plot_Widget()
304         plot.show()
305         sys.exit(app.exec_())
306
307     #=====
308     # END Program
309     #=====

```

A.2 GUI Listing

A.2.1 Pulse.ui Listing

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <ui version="4.0">
3      <class>Form</class>
4      <widget class="QWidget" name="Form">
5          <property name="geometry">
6              <rect>
7                  <x>0</x>
8                  <y>0</y>
9                  <width>1000</width>
10                 <height>679</height>
11             </rect>
12         </property>
13         <property name="minimumSize">
14             <size>
15                 <width>855</width>
16                 <height>539</height>

```

```

17     </size>
18 </property>
19 <property name="maximumSize">
20     <size>
21         <width>1000</width>
22         <height>1200</height>
23     </size>
24 </property>
25 <property name="windowTitle">
26     <string>Form</string>
27 </property>
28 <widget class="QWidget" name="layoutWidget">
29     <property name="geometry">
30         <rect>
31             <x>20</x>
32             <y>280</y>
33             <width>231</width>
34             <height>71</height>
35         </rect>
36     </property>
37     <layout class="QGridLayout" name="gridLayout_2">
38         <item row="0" column="0">
39             <widget class="QLabel" name="label_8">
40                 <property name="text">
41                     <string>&lt;!DOCTYPE HTML PUBLIC &quot;
42                         /strict.dtd&quot;&gt;
43                     &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;
44                         qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text /
45                         css&quot;&gt;
46                         p, li { white-space: pre-wrap; }
47                     &lt; /style&gt;&lt; /head&gt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
48                         style:normal;&quot;&gt;
49                     &lt; p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
50                         indent:0px;&quot;&gt;&lt;span style=&quot; font-weight:600;&quot;&gt;Amplitude (V)&lt; /span&gt;&lt; /p&gt;&lt; /body&gt;&lt; /html&gt;&lt; /string>
51                 </property>
52             </widget>
53         </item>
54         <item row="0" column="1">
55             <widget class="QLineEdit" name="V_max">
56                 <property name="toolTip">
57                     <string>Amplitude of input pulse</string>
58                 </property>
59                 <property name="text">
60                     <string>l</string>
61                 </property>
62             </widget>
63         </item>
64         <item row="1" column="0">
65             <widget class="QLabel" name="label_9">
66                 <property name="text">
67                     <string>&lt;!DOCTYPE HTML PUBLIC &quot;
68                         /strict.dtd&quot;&gt;
69                     &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;
70                         qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text /
71                         css&quot;&gt;
72                         p, li { white-space: pre-wrap; }
73                     &lt; /style&gt;&lt; /head&gt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
74                         style:normal;&quot;&gt;
75                     &lt; p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
76                         indent:0px;&quot;&gt;&lt;span style=&quot; font-weight:600;&quot;&gt;Pulse Width (ms)&lt; /span&gt;&lt; /p&gt;&lt; /body&gt;&lt; /html&gt;&lt; /string>

```

```

67         </property>
68     </widget>
69 </item>
70 <item row="1" column="1">
71     <widget class="QLineEdit" name="Pulse_width">
72         <property name="toolTip">
73             <string>Pulse width of input pulse
74 in milliseconds (ms)</string>
75         </property>
76         <property name="text">
77             <string>0.001</string>
78         </property>
79     </widget>
80 </item>
81 </layout>
82 </widget>
83 <widget class="QPushButton" name="VPM">
84     <property name="geometry">
85         <rect>
86             <x>10</x>
87             <y>600</y>
88             <width>111</width>
89             <height>28</height>
90         </rect>
91     </property>
92     <property name="toolTip">
93         <string>Plots Voltage Propagation Model (VPM) Pulse Shape</string>
94     </property>
95     <property name="text">
96         <string>Plot &VPM</string>
97     </property>
98 </widget>
99 <widget class="MPL_Widget" name="plotWidget" native="true">
100     <property name="geometry">
101         <rect>
102             <x>270</x>
103             <y>20</y>
104             <width>711</width>
105             <height>641</height>
106         </rect>
107     </property>
108     <property name="toolTip">
109         <string>matplotlib plot area</string>
110     </property>
111 </widget>
112 <widget class="QLabel" name="label_11">
113     <property name="geometry">
114         <rect>
115             <x>120</x>
116             <y>560</y>
117             <width>62</width>
118             <height>18</height>
119         </rect>
120     </property>
121     <property name="text">
122         <string>m</string>
123     </property>
124 </widget>
125 <widget class="QPushButton" name="Close">
126     <property name="geometry">

```



```

127     <rect>
128         <x>130</x>
129         <y>640</y>
130         <width>111</width>
131         <height>28</height>
132     </rect>
133 </property>
134 <property name="toolTip">
135     <string>Close the application</string>
136 </property>
137 <property name="text">
138     <string>&amp; Close</string>
139 </property>
140 </widget>
141 <widget class="Line" name="line_2">
142     <property name="geometry">
143         <rect>
144             <x>250</x>
145             <y>0</y>
146             <width>20</width>
147             <height>661</height>
148         </rect>
149     </property>
150     <property name="orientation">
151         <enum>Qt::Vertical</enum>
152     </property>
153 </widget>
154 <widget class="Line" name="line">
155     <property name="geometry">
156         <rect>
157             <x>0</x>
158             <y>250</y>
159             <width>251</width>
160             <height>16</height>
161         </rect>
162     </property>
163     <property name="orientation">
164         <enum>Qt::Horizontal</enum>
165     </property>
166 </widget>
167 <widget class="QWidget" name="layoutWidget_2">
168     <property name="geometry">
169         <rect>
170             <x>20</x>
171             <y>50</y>
172             <width>231</width>
173             <height>204</height>
174         </rect>
175     </property>
176     <layout class="QGridLayout" name="gridLayout">
177         <property name="sizeConstraint">
178             <enum>QLayout::SetFixedSize</enum>
179         </property>
180         <item row="0" column="0">
181             <widget class="QLabel" name="label">
182                 <property name="text">
183                     <string>&lt;!DOCTYPE HTML PUBLIC &quot; -//W3C//DTD HTML 4.0//EN&quot; &quot; http://www.w3.org/TR/REC-html40
                        /strict.dtd&quot;&gt;
184 &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
                        css&quot;&gt;

```

```

185 p, li { white-space: pre-wrap; }
186 <lt;/style>&lt;&lt;/head>&lt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
    style:normal;&quot;&gt;&gt;
187 &lt;lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
    indent:0px;&quot;&gt;&lt;&lt;span style=&quot; font-weight:600;&quot;&gt;&lt;R (ohm)&lt;/span>&lt;&lt;/p>&lt;&lt;/body>
    &lt;&lt;/html>&lt;&lt;/string>
188 </property>
189 </widget>
190 </item>
191 <item row="1" column="0">
192 <widget class="QLabel" name="label_7">
193 <property name="text">
194 <string>&lt;&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40
    /strict.dtd&quot;&gt;&gt;
195 &lt;&lt;html>&lt;&lt;head>&lt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;&lt;style type=&quot;text/
    css&quot;&gt;&gt;
196 p, li { white-space: pre-wrap; }
197 &lt;lt;/style>&lt;&lt;/head>&lt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
    style:normal;&quot;&gt;&gt;
198 &lt;lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
    indent:0px;&quot;&gt;&lt;&lt;span style=&quot; font-weight:600;&quot;&gt;&lt;L (H)&lt;/span>&lt;&lt;/p>&lt;&lt;/body>&lt;
    &lt;&lt;/html>&lt;&lt;/string>
199 </property>
200 </widget>
201 </item>
202 <item row="2" column="0">
203 <widget class="QLabel" name="label_3">
204 <property name="text">
205 <string>&lt;&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40
    /strict.dtd&quot;&gt;&gt;
206 &lt;&lt;html>&lt;&lt;head>&lt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;&lt;style type=&quot;text/
    css&quot;&gt;&gt;
207 p, li { white-space: pre-wrap; }
208 &lt;lt;/style>&lt;&lt;/head>&lt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
    style:normal;&quot;&gt;&gt;
209 &lt;lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
    indent:0px;&quot;&gt;&lt;&lt;span style=&quot; font-weight:600;&quot;&gt;&lt;C (C)&lt;/span>&lt;&lt;/p>&lt;&lt;/body>&lt;
    &lt;&lt;/html>&lt;&lt;/string>
210 </property>
211 </widget>
212 </item>
213 <item row="3" column="0">
214 <widget class="QLabel" name="label_4">
215 <property name="text">
216 <string>&lt;&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40
    /strict.dtd&quot;&gt;&gt;
217 &lt;&lt;html>&lt;&lt;head>&lt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;&lt;style type=&quot;text/
    css&quot;&gt;&gt;
218 p, li { white-space: pre-wrap; }
219 &lt;lt;/style>&lt;&lt;/head>&lt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
    style:normal;&quot;&gt;&gt;
220 &lt;lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
    indent:0px;&quot;&gt;&lt;&lt;span style=&quot; font-weight:600;&quot;&gt;&lt;G (mho)&lt;/span>&lt;&lt;/p>&lt;&lt;/body>
    &lt;&lt;/html>&lt;&lt;/string>
221 </property>
222 </widget>
223 </item>
224 <item row="4" column="0">
225 <widget class="QLabel" name="label_5">
226 <property name="text">

```

```

227         <string>&lt;!DOCTYPE HTML PUBLIC &quot;--//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40
           /strict.dtd&quot;&gt;
228 &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
           css&quot;&gt;
229 p, li { white-space: pre-wrap; }
230 &lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family: 'Sans'; font-size:10pt; font-weight:400; font-
           style:normal;&quot;&gt;
231 &lt;p align=&quot;justify&quot; style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
           -qt-block-indent:0; text-indent:0px;&quot;&gt;&lt;span style=&quot; font-weight:600;&quot;&gt;Average tap&lt
           /span&gt;&lt;/p&gt;
232 &lt;p align=&quot;justify&quot; style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
           -qt-block-indent:0; text-indent:0px;&quot;&gt;&lt;span style=&quot; font-weight:600;&quot;&gt;distance (m)&
           lt;/span&gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>
233 </property>
234 </widget>
235 </item>
236 <item row="5" column="0">
237 <widget class="QLabel" name="label_6">
238 <property name="text">
239 <string>&lt;!DOCTYPE HTML PUBLIC &quot;--//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40
           /strict.dtd&quot;&gt;
240 &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
           css&quot;&gt;
241 p, li { white-space: pre-wrap; }
242 &lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family: 'Sans'; font-size:10pt; font-weight:400; font-
           style:normal;&quot;&gt;
243 &lt;p align=&quot;justify&quot; style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
           -qt-block-indent:0; text-indent:0px;&quot;&gt;&lt;span style=&quot; font-weight:600;&quot;&gt;Length of &lt
           /span&gt;&lt;/p&gt;
244 &lt;p align=&quot;justify&quot; style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px;
           -qt-block-indent:0; text-indent:0px;&quot;&gt;&lt;span style=&quot; font-weight:600;&quot;&gt;tx. line (m)&
           lt;/span&gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>
245 </property>
246 </widget>
247 </item>
248 <item row="0" column="1">
249 <widget class="QLineEdit" name="R">
250 <property name="maximumSize">
251 <size>
252 <width>70</width>
253 <height>26</height>
254 </size>
255 </property>
256 <property name="toolTip">
257 <string>Characteristic Resistance of the transmission line</string>
258 </property>
259 <property name="text">
260 <string>0.35</string>
261 </property>
262 </widget>
263 </item>
264 <item row="1" column="1">
265 <widget class="QLineEdit" name="L">
266 <property name="maximumSize">
267 <size>
268 <width>70</width>
269 <height>26</height>
270 </size>
271 </property>
272 <property name="toolTip">

```

```

273         <string>Characteristic Inductance of
274 the transmission line</string>
275     </property>
276     <property name="text">
277         <string>0.60e-6</string>
278     </property>
279 </widget>
280 </item>
281 <item row="2" column="1">
282     <widget class="QLineEdit" name="C">
283         <property name="maximumSize">
284             <size>
285                 <width>70</width>
286                 <height>26</height>
287             </size>
288         </property>
289         <property name="toolTip">
290             <string>Characteristic Capacitance of
291 the transmission line</string>
292         </property>
293         <property name="text">
294             <string>60.0e-9</string>
295         </property>
296     </widget>
297 </item>
298 <item row="3" column="1">
299     <widget class="QLineEdit" name="G">
300         <property name="maximumSize">
301             <size>
302                 <width>70</width>
303                 <height>26</height>
304             </size>
305         </property>
306         <property name="toolTip">
307             <string>Characteristic Conductance of
308 the transmission line</string>
309         </property>
310         <property name="text">
311             <string>0.19e-3</string>
312         </property>
313     </widget>
314 </item>
315 <item row="4" column="1">
316     <widget class="QLineEdit" name="N">
317         <property name="maximumSize">
318             <size>
319                 <width>70</width>
320                 <height>26</height>
321             </size>
322         </property>
323         <property name="toolTip">
324             <string>Average distance between the transission
325 line junction/node points</string>
326         </property>
327         <property name="text">
328             <string>50</string>
329         </property>
330     </widget>
331 </item>
332 <item row="5" column="1">

```

```

333     <widget class="QLineEdit" name="D">
334         <property name="maximumSize">
335             <size>
336                 <width>70</width>
337                 <height>26</height>
338             </size>
339         </property>
340         <property name="toolTip">
341             <string>Total length of the
342 transmission line</string>
343         </property>
344         <property name="text">
345             <string>100</string>
346         </property>
347     </widget>
348 </item>
349 </layout>
350 </widget>
351 <widget class="Line" name="line_3">
352     <property name="geometry">
353         <rect>
354             <x>0</x>
355             <y>350</y>
356             <width>251</width>
357             <height>16</height>
358         </rect>
359     </property>
360     <property name="orientation">
361         <enum>Qt::Horizontal</enum>
362     </property>
363 </widget>
364 <widget class="Line" name="line_4">
365     <property name="geometry">
366         <rect>
367             <x>0</x>
368             <y>580</y>
369             <width>251</width>
370             <height>16</height>
371         </rect>
372     </property>
373     <property name="orientation">
374         <enum>Qt::Horizontal</enum>
375     </property>
376 </widget>
377 <widget class="QLineEdit" name="x_position">
378     <property name="geometry">
379         <rect>
380             <x>40</x>
381             <y>550</y>
382             <width>81</width>
383             <height>26</height>
384         </rect>
385     </property>
386     <property name="toolTip">
387         <string>Point of interest on the
388 transmission line .
389 input should be in metres (m)
390 form signal feed point</string>
391     </property>
392     <property name="text">

```

```

393     <string>50</string>
394 </property>
395 </widget>
396 <widget class="QLabel" name="label_2">
397     <property name="geometry">
398         <rect>
399             <x>20</x>
400             <y>7</y>
401             <width>231</width>
402             <height>31</height>
403         </rect>
404     </property>
405     <property name="text">
406         <string>&lt;!DOCTYPE HTML PUBLIC &quot;--//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/
            strict.dtd&quot;&gt;
407 &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;richtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
            css&quot;&gt;
408 p, li { white-space: pre-wrap; }
409 &lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
            style:normal;&quot;&gt;
410 &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
            indent:0px;&quot;&gt;&lt;span style=&quot; color:#0000ff;&quot;&gt;R, L, C and G are the characteristic &lt;/
            span&gt;&lt;/p&gt;
411 &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
            indent:0px;&quot;&gt;&lt;span style=&quot; color:#0000ff;&quot;&gt;values of the Transmission line&lt;/span&
            &gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>
412     </property>
413 </widget>
414 <widget class="QLabel" name="label_12">
415     <property name="geometry">
416         <rect>
417             <x>20</x>
418             <y>490</y>
419             <width>191</width>
420             <height>51</height>
421         </rect>
422     </property>
423     <property name="text">
424         <string>&lt;!DOCTYPE HTML PUBLIC &quot;--//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/
            strict.dtd&quot;&gt;
425 &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;richtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
            css&quot;&gt;
426 p, li { white-space: pre-wrap; }
427 &lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
            style:normal;&quot;&gt;
428 &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
            indent:0px;&quot;&gt;&lt;span style=&quot; color:#0000ff;&quot;&gt;Position of Interest &lt;/span&gt;&lt;/p&
            &gt;
429 &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
            indent:0px;&quot;&gt;&lt;span style=&quot; color:#0000ff;&quot;&gt;( Distance from signal source&lt;/span&gt
            &lt;/p&gt;
430 &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
            indent:0px;&quot;&gt;&lt;span style=&quot; color:#0000ff;&quot;&gt;that pulse is viewed)&lt;/span&gt;&lt;/p&
            &gt;&lt;/body&gt;&lt;/html&gt;</string>
431     </property>
432 </widget>
433 <widget class="QLabel" name="label_10">
434     <property name="geometry">
435         <rect>
436             <x>20</x>

```



```

437     <y>260</y>
438     <width>161</width>
439     <height>21</height>
440     </rect>
441     </property>
442     <property name="text">
443         <string>&lt;!DOCTYPE HTML PUBLIC &quot; -//W3C//DTD HTML 4.0//EN&quot; &quot; http://www.w3.org/TR/REC-html40/
            strict.dtd&quot;&gt;
444     &lt;html&gt;&lt;head&gt;&lt;meta name=&quot; qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
            css&quot;&gt;
445     p, li { white-space: pre-wrap; }
446     &lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
            style:normal;&quot;&gt;
447     &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
            indent:0px;&quot;&gt;&lt;span style=&quot; color:#0000ff;&quot;&gt;Input Pulse Properties&lt;/span&gt;&lt;p&
            &gt;&lt;/body&gt;&lt;/html&gt;&lt;/string>
448     </property>
449     </widget>
450     <widget class="Line" name="line_5">
451         <property name="geometry">
452             <rect>
453                 <x>0</x>
454                 <y>670</y>
455                 <width>991</width>
456                 <height>20</height>
457             </rect>
458         </property>
459         <property name="orientation">
460             <enum>Qt::Horizontal</enum>
461         </property>
462     </widget>
463     <widget class="Line" name="line_6">
464         <property name="geometry">
465             <rect>
466                 <x>260</x>
467                 <y>0</y>
468                 <width>731</width>
469                 <height>16</height>
470             </rect>
471         </property>
472         <property name="orientation">
473             <enum>Qt::Horizontal</enum>
474         </property>
475     </widget>
476     <widget class="QPushButton" name="IPM">
477         <property name="geometry">
478             <rect>
479                 <x>130</x>
480                 <y>600</y>
481                 <width>111</width>
482                 <height>28</height>
483             </rect>
484         </property>
485         <property name="toolTip">
486             <string>Plot Current Propagation Model (IPM) pulse shape</string>
487         </property>
488         <property name="text">
489             <string>Plot &IPM</string>
490         </property>
491     </widget>

```



```

492     <widget class="QWidget" name="layoutWidget_3">
493     <property name="geometry">
494     <rect>
495     <x>20</x>
496     <y>380</y>
497     <width>231</width>
498     <height>92</height>
499     </rect>
500     </property>
501     <layout class="QGridLayout" name="gridLayout_3">
502     <property name="sizeConstraint">
503     <enum>QLayout::SetFixedSize</enum>
504     </property>
505     <item row="0" column="0">
506     <widget class="QLabel" name="label_13">
507     <property name="text">
508     <string>&lt;!DOCTYPE HTML PUBLIC &quot;--//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40
509     /strict.dtd&quot;&gt;
510     &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
511     css&quot;&gt;
512     p, li { white-space: pre-wrap; }
513     &lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
514     style:normal;&quot;&gt;
515     &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
516     indent:0px;&quot;&gt;&lt;span style=&quot; font-weight:600;&quot;&gt;Rr (ohm)&lt;/span&gt;&lt;/p&gt;&lt;/body&
517     &gt;&lt;/html&gt;</string>
518     </property>
519     </widget>
520     </item>
521     <item row="1" column="0">
522     <widget class="QLabel" name="label_14">
523     <property name="text">
524     <string>&lt;!DOCTYPE HTML PUBLIC &quot;--//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40
525     /strict.dtd&quot;&gt;
526     &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
527     css&quot;&gt;
528     p, li { white-space: pre-wrap; }
529     &lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
530     style:normal;&quot;&gt;
531     &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
532     indent:0px;&quot;&gt;&lt;span style=&quot; font-weight:600;&quot;&gt;Lr (H)&lt;/span&gt;&lt;/p&gt;&lt;/body&
533     &gt;&lt;/html&gt;</string>
534     </property>
535     </widget>
536     </item>
537     <item row="2" column="0">
538     <widget class="QLabel" name="label_15">
539     <property name="text">
540     <string>&lt;!DOCTYPE HTML PUBLIC &quot;--//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40
541     /strict.dtd&quot;&gt;
542     &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
543     css&quot;&gt;
544     p, li { white-space: pre-wrap; }
545     &lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
546     style:normal;&quot;&gt;
547     &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
548     indent:0px;&quot;&gt;&lt;span style=&quot; font-weight:600;&quot;&gt;Cr (C)&lt;/span&gt;&lt;/p&gt;&lt;/body&
549     &gt;&lt;/html&gt;</string>
550     </property>
551     </widget>

```

```

537     </item>
538     <item row="0" column="1">
539         <widget class="QLineEdit" name="Rr">
540             <property name="maximumSize">
541                 <size>
542                     <width>70</width>
543                     <height>26</height>
544                 </size>
545             </property>
546             <property name="toolTip">
547                 <string>Resistance of the tap load</string>
548             </property>
549             <property name="text">
550                 <string>1e3</string>
551             </property>
552         </widget>
553     </item>
554     <item row="1" column="1">
555         <widget class="QLineEdit" name="Lr">
556             <property name="maximumSize">
557                 <size>
558                     <width>70</width>
559                     <height>26</height>
560                 </size>
561             </property>
562             <property name="toolTip">
563                 <string>Inductance of the tap load</string>
564             </property>
565             <property name="text">
566                 <string>47e-9</string>
567             </property>
568         </widget>
569     </item>
570     <item row="2" column="1">
571         <widget class="QLineEdit" name="Cr">
572             <property name="maximumSize">
573                 <size>
574                     <width>70</width>
575                     <height>26</height>
576                 </size>
577             </property>
578             <property name="toolTip">
579                 <string>Capacitance of the tap load</string>
580             </property>
581             <property name="text">
582                 <string>1.3e-3</string>
583             </property>
584         </widget>
585     </item>
586 </layout>
587 </widget>
588 <widget class="Line" name="line_7">
589     <property name="geometry">
590         <rect>
591             <x>0</x>
592             <y>480</y>
593             <width>251</width>
594             <height>16</height>
595         </rect>
596     </property>

```

```

597     <property name="orientation">
598         <enum>Qt::Horizontal</enum>
599     </property>
600 </widget>
601 <widget class="QLabel" name="label_16">
602     <property name="geometry">
603         <rect>
604             <x>20</x>
605             <y>360</y>
606             <width>191</width>
607             <height>21</height>
608         </rect>
609     </property>
610     <property name="text">
611         <string>&lt;!DOCTYPE HTML PUBLIC &quot;--//W3C//DTD HTML 4.0//EN&quot; &quot;http://www.w3.org/TR/REC-html40/
            strict.dtd&quot;&gt;
612 &lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;l&quot; /&gt;&lt;style type=&quot;text/
            css&quot;&gt;
613 p, li { white-space: pre-wrap; }
614 &lt;/style&gt;&lt;head&gt;&lt;body style=&quot; font-family:'Sans'; font-size:10pt; font-weight:400; font-
            style:normal;&quot;&gt;
615 &lt;p style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-
            indent:0px;&quot;&gt;&lt;span style=&quot; color:#0000ff;&quot;&gt;Load Parameters&lt;/span&gt;&lt;p&gt;&lt;
            /body&gt;&lt;/html&gt;&lt;/string>
616     </property>
617 </widget>
618 <widget class="Line" name="line_8">
619     <property name="geometry">
620         <rect>
621             <x>980</x>
622             <y>10</y>
623             <width>20</width>
624             <height>661</height>
625         </rect>
626     </property>
627     <property name="orientation">
628         <enum>Qt::Vertical</enum>
629     </property>
630 </widget>
631 <widget class="QPushButton" name="Clear">
632     <property name="geometry">
633         <rect>
634             <x>10</x>
635             <y>640</y>
636             <width>111</width>
637             <height>27</height>
638         </rect>
639     </property>
640     <property name="toolTip">
641         <string>Clears the matplotlib plot area</string>
642     </property>
643     <property name="text">
644         <string>Clear Plot Area</string>
645     </property>
646 </widget>
647 </widget>
648 <customwidgets>
649     <customwidget>
650         <class>MPL_Widget</class>
651         <extends>QWidget</extends>

```

```

652     <header>mpl_pyqt4_widget</header>
653     <container>1</container>
654     </customwidget>
655 </customwidgets>
656 <tabstops>
657     <tabstop>R</tabstop>
658     <tabstop>L</tabstop>
659     <tabstop>C</tabstop>
660     <tabstop>G</tabstop>
661     <tabstop>N</tabstop>
662     <tabstop>D</tabstop>
663     <tabstop>V_max</tabstop>
664     <tabstop>Pulse_width</tabstop>
665     <tabstop>x_position</tabstop>
666     <tabstop>VPM</tabstop>
667     <tabstop>Close</tabstop>
668 </tabstops>
669 <resources />
670 <connections>
671     <connection>
672         <sender>Close</sender>
673         <signal>clicked()</signal>
674         <receiver>Form</receiver>
675         <slot>close()</slot>
676     <hints>
677         <hint type="sourcelabel">
678             <x>240</x>
679             <y>640</y>
680         </hint>
681         <hint type="destinationlabel">
682             <x>229</x>
683             <y>455</y>
684         </hint>
685     </hints>
686 </connection>
687 </connections>
688 </ui>

```

A.2.2 GUI Python Listing

```

1  # -*- coding: utf-8 -*-
2
3  # Form implementation generated from reading ui file 'Pulse.ui'
4  #
5  # Created: Tue Sep  7 09:42:25 2010
6  #       by: PyQt4 UI code generator 4.7.2
7  #
8  # WARNING! All changes made in this file will be lost!
9
10 from PyQt4 import QtCore, QtGui
11
12 class Ui_Form(object):
13     def setupUi(self, Form):
14         Form.setObjectName("Form")
15         Form.resize(1000, 679)
16         Form.setMinimumSize(QtCore.QSize(855, 539))
17         Form.setMaximumSize(QtCore.QSize(1000, 1200))
18         self.layoutWidget = QtGui.QWidget(Form)

```

```

19     self.layoutWidget.setGeometry(QRect(20, 280, 231, 71))
20     self.layoutWidget.setObjectName("layoutWidget")
21     self.gridLayout_2 = QtGui.QGridLayout(self.layoutWidget)
22     self.gridLayout_2.setObjectName("gridLayout_2")
23     self.label_8 = QtGui.QLabel(self.layoutWidget)
24     self.label_8.setObjectName("label_8")
25     self.gridLayout_2.addWidget(self.label_8, 0, 0, 1, 1)
26     self.V_max = QtGui.QLineEdit(self.layoutWidget)
27     self.V_max.setObjectName("V_max")
28     self.gridLayout_2.addWidget(self.V_max, 0, 1, 1, 1)
29     self.label_9 = QtGui.QLabel(self.layoutWidget)
30     self.label_9.setObjectName("label_9")
31     self.gridLayout_2.addWidget(self.label_9, 1, 0, 1, 1)
32     self.Pulse_width = QtGui.QLineEdit(self.layoutWidget)
33     self.Pulse_width.setObjectName("Pulse_width")
34     self.gridLayout_2.addWidget(self.Pulse_width, 1, 1, 1, 1)
35     self.VPM = QtGui.QPushButton(Form)
36     self.VPM.setGeometry(QRect(10, 600, 111, 28))
37     self.VPM.setObjectName("VPM")
38     self.plotWidget = MPL_Widget(Form)
39     self.plotWidget.setGeometry(QRect(270, 20, 711, 641))
40     self.plotWidget.setObjectName("plotWidget")
41     self.label_11 = QtGui.QLabel(Form)
42     self.label_11.setGeometry(QRect(120, 560, 62, 18))
43     self.label_11.setObjectName("label_11")
44     self.Close = QtGui.QPushButton(Form)
45     self.Close.setGeometry(QRect(130, 640, 111, 28))
46     self.Close.setObjectName("Close")
47     self.line_2 = QtGui.QFrame(Form)
48     self.line_2.setGeometry(QRect(250, 0, 20, 661))
49     self.line_2.setFrameShape(QtGui.QFrame.VLine)
50     self.line_2.setFrameShadow(QtGui.QFrame.Sunken)
51     self.line_2.setObjectName("line_2")
52     self.line = QtGui.QFrame(Form)
53     self.line.setGeometry(QRect(0, 250, 251, 16))
54     self.line.setFrameShape(QtGui.QFrame.HLine)
55     self.line.setFrameShadow(QtGui.QFrame.Sunken)
56     self.line.setObjectName("line")
57     self.layoutWidget_2 = QtGui.QWidget(Form)
58     self.layoutWidget_2.setGeometry(QRect(20, 50, 231, 204))
59     self.layoutWidget_2.setObjectName("layoutWidget_2")
60     self.gridLayout = QtGui.QGridLayout(self.layoutWidget_2)
61     self.gridLayout.setSizeConstraint(QtGui.QLayout.SetFixedSize)
62     self.gridLayout.setObjectName("gridLayout")
63     self.label = QtGui.QLabel(self.layoutWidget_2)
64     self.label.setObjectName("label")
65     self.gridLayout.addWidget(self.label, 0, 0, 1, 1)
66     self.label_7 = QtGui.QLabel(self.layoutWidget_2)
67     self.label_7.setObjectName("label_7")
68     self.gridLayout.addWidget(self.label_7, 1, 0, 1, 1)
69     self.label_3 = QtGui.QLabel(self.layoutWidget_2)
70     self.label_3.setObjectName("label_3")
71     self.gridLayout.addWidget(self.label_3, 2, 0, 1, 1)
72     self.label_4 = QtGui.QLabel(self.layoutWidget_2)
73     self.label_4.setObjectName("label_4")
74     self.gridLayout.addWidget(self.label_4, 3, 0, 1, 1)
75     self.label_5 = QtGui.QLabel(self.layoutWidget_2)
76     self.label_5.setObjectName("label_5")
77     self.gridLayout.addWidget(self.label_5, 4, 0, 1, 1)
78     self.label_6 = QtGui.QLabel(self.layoutWidget_2)

```

```

79     self.label_6.setObjectName("label_6")
80     self.gridLayout.addWidget(self.label_6, 5, 0, 1, 1)
81     self.R = QtGui.QLineEdit(self.layoutWidget_2)
82     self.R.setMaximumSize(QtCore.QSize(70, 26))
83     self.R.setObjectName("R")
84     self.gridLayout.addWidget(self.R, 0, 1, 1, 1)
85     self.L = QtGui.QLineEdit(self.layoutWidget_2)
86     self.L.setMaximumSize(QtCore.QSize(70, 26))
87     self.L.setObjectName("L")
88     self.gridLayout.addWidget(self.L, 1, 1, 1, 1)
89     self.C = QtGui.QLineEdit(self.layoutWidget_2)
90     self.C.setMaximumSize(QtCore.QSize(70, 26))
91     self.C.setObjectName("C")
92     self.gridLayout.addWidget(self.C, 2, 1, 1, 1)
93     self.G = QtGui.QLineEdit(self.layoutWidget_2)
94     self.G.setMaximumSize(QtCore.QSize(70, 26))
95     self.G.setObjectName("G")
96     self.gridLayout.addWidget(self.G, 3, 1, 1, 1)
97     self.N = QtGui.QLineEdit(self.layoutWidget_2)
98     self.N.setMaximumSize(QtCore.QSize(70, 26))
99     self.N.setObjectName("N")
100    self.gridLayout.addWidget(self.N, 4, 1, 1, 1)
101    self.D = QtGui.QLineEdit(self.layoutWidget_2)
102    self.D.setMaximumSize(QtCore.QSize(70, 26))
103    self.D.setObjectName("D")
104    self.gridLayout.addWidget(self.D, 5, 1, 1, 1)
105    self.line_3 = QtGui.QFrame(Form)
106    self.line_3.setGeometry(QtCore.QRect(0, 350, 251, 16))
107    self.line_3.setFrameShape(QtGui.QFrame.HLine)
108    self.line_3.setFrameShadow(QtGui.QFrame.Sunken)
109    self.line_3.setObjectName("line_3")
110    self.line_4 = QtGui.QFrame(Form)
111    self.line_4.setGeometry(QtCore.QRect(0, 580, 251, 16))
112    self.line_4.setFrameShape(QtGui.QFrame.HLine)
113    self.line_4.setFrameShadow(QtGui.QFrame.Sunken)
114    self.line_4.setObjectName("line_4")
115    self.x_position = QtGui.QLineEdit(Form)
116    self.x_position.setGeometry(QtCore.QRect(40, 550, 81, 26))
117    self.x_position.setObjectName("x_position")
118    self.label_2 = QtGui.QLabel(Form)
119    self.label_2.setGeometry(QtCore.QRect(20, 7, 231, 31))
120    self.label_2.setObjectName("label_2")
121    self.label_12 = QtGui.QLabel(Form)
122    self.label_12.setGeometry(QtCore.QRect(20, 490, 191, 51))
123    self.label_12.setObjectName("label_12")
124    self.label_10 = QtGui.QLabel(Form)
125    self.label_10.setGeometry(QtCore.QRect(20, 260, 161, 21))
126    self.label_10.setObjectName("label_10")
127    self.line_5 = QtGui.QFrame(Form)
128    self.line_5.setGeometry(QtCore.QRect(0, 670, 991, 20))
129    self.line_5.setFrameShape(QtGui.QFrame.HLine)
130    self.line_5.setFrameShadow(QtGui.QFrame.Sunken)
131    self.line_5.setObjectName("line_5")
132    self.line_6 = QtGui.QFrame(Form)
133    self.line_6.setGeometry(QtCore.QRect(260, 0, 731, 16))
134    self.line_6.setFrameShape(QtGui.QFrame.HLine)
135    self.line_6.setFrameShadow(QtGui.QFrame.Sunken)
136    self.line_6.setObjectName("line_6")
137    self.IPM = QtGui.QPushButton(Form)
138    self.IPM.setGeometry(QtCore.QRect(130, 600, 111, 28))

```



```

139         self.IPM.setObjectName("IPM")
140         self.layoutWidget_3 = QtGui.QWidget(Form)
141         self.layoutWidget_3.setGeometry(QtCore.QRect(20, 380, 231, 92))
142         self.layoutWidget_3.setObjectName("layoutWidget_3")
143         self.gridLayout_3 = QtGui.QGridLayout(self.layoutWidget_3)
144         self.gridLayout_3.setSizeConstraint(QtGui.QLayout.SetFixedSize)
145         self.gridLayout_3.setObjectName("gridLayout_3")
146         self.label_13 = QtGui.QLabel(self.layoutWidget_3)
147         self.label_13.setObjectName("label_13")
148         self.gridLayout_3.addWidget(self.label_13, 0, 0, 1, 1)
149         self.label_14 = QtGui.QLabel(self.layoutWidget_3)
150         self.label_14.setObjectName("label_14")
151         self.gridLayout_3.addWidget(self.label_14, 1, 0, 1, 1)
152         self.label_15 = QtGui.QLabel(self.layoutWidget_3)
153         self.label_15.setObjectName("label_15")
154         self.gridLayout_3.addWidget(self.label_15, 2, 0, 1, 1)
155         self.Rr = QtGui.QLineEdit(self.layoutWidget_3)
156         self.Rr.setMaximumSize(QtCore.QSize(70, 26))
157         self.Rr.setObjectName("Rr")
158         self.gridLayout_3.addWidget(self.Rr, 0, 1, 1, 1)
159         self.Lr = QtGui.QLineEdit(self.layoutWidget_3)
160         self.Lr.setMaximumSize(QtCore.QSize(70, 26))
161         self.Lr.setObjectName("Lr")
162         self.gridLayout_3.addWidget(self.Lr, 1, 1, 1, 1)
163         self.Cr = QtGui.QLineEdit(self.layoutWidget_3)
164         self.Cr.setMaximumSize(QtCore.QSize(70, 26))
165         self.Cr.setObjectName("Cr")
166         self.gridLayout_3.addWidget(self.Cr, 2, 1, 1, 1)
167         self.line_7 = QtGui.QFrame(Form)
168         self.line_7.setGeometry(QtCore.QRect(0, 480, 251, 16))
169         self.line_7.setFrameShape(QtGui.QFrame.HLine)
170         self.line_7.setFrameShadow(QtGui.QFrame.Sunken)
171         self.line_7.setObjectName("line_7")
172         self.label_16 = QtGui.QLabel(Form)
173         self.label_16.setGeometry(QtCore.QRect(20, 360, 191, 21))
174         self.label_16.setObjectName("label_16")
175         self.line_8 = QtGui.QFrame(Form)
176         self.line_8.setGeometry(QtCore.QRect(980, 10, 20, 661))
177         self.line_8.setFrameShape(QtGui.QFrame.VLine)
178         self.line_8.setFrameShadow(QtGui.QFrame.Sunken)
179         self.line_8.setObjectName("line_8")
180         self.Clear = QtGui.QPushButton(Form)
181         self.Clear.setGeometry(QtCore.QRect(10, 640, 111, 27))
182         self.Clear.setObjectName("Clear")
183
184         self.retranslateUi(Form)
185         QtCore.QObject.connect(self.Close, QtCore.SIGNAL("clicked()"), Form.close)
186         QtCore.QMetaObject.connectSlotsByName(Form)
187         Form.setTabOrder(self.R, self.L)
188         Form.setTabOrder(self.L, self.C)
189         Form.setTabOrder(self.C, self.G)
190         Form.setTabOrder(self.G, self.N)
191         Form.setTabOrder(self.N, self.D)
192         Form.setTabOrder(self.D, self.V_max)
193         Form.setTabOrder(self.V_max, self.Pulse_width)
194         Form.setTabOrder(self.Pulse_width, self.x_position)
195         Form.setTabOrder(self.x_position, self.VPM)
196         Form.setTabOrder(self.VPM, self.Close)
197
198     def retranslateUi(self, Form):

```

```

199         Form.setWindowTitle(QtGui.QApplication.translate("Form", "Pulse_Propagation_on_Multi-Access_Wireline_
                Channels", None, QtGui.QApplication.UnicodeUTF8))
200         self.label_8.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
                EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
201         "<html><head><meta_name=\\_ qrichtext\\_<content=\\_1\\_</><style_type=\\_ text/css\\_>\\n"
202         "p_{_white-space:_pre-wrap;_}\\n"
203         "</style></head><body_style=\\_<font-family:_\\_ Sans\\_';<font-size:10pt;<font-weight:400;<font-style:normal;\\_>\\n"
204         "<p_style=\\_<margin-top:0px;<margin-bottom:0px;<margin-left:0px;<margin-right:0px;<-qt-block-indent:0;<text-indent
                :0px;\\_><span_style=\\_<font-weight:600;\\_>Amplitude_(V)</span></p></body></html>", None, QtGui.QApplication.
                UnicodeUTF8))
205         self.V_max.setToolTip(QtGui.QApplication.translate("Form", "Amplitude_of_input_pulse", None, QtGui.
                QApplication.UnicodeUTF8))
206         self.V_max.setText(QtGui.QApplication.translate("Form", "1", None, QtGui.QApplication.UnicodeUTF8))
207         self.label_9.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
                EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
208         "<html><head><meta_name=\\_ qrichtext\\_<content=\\_1\\_</><style_type=\\_ text/css\\_>\\n"
209         "p_{_white-space:_pre-wrap;_}\\n"
210         "</style></head><body_style=\\_<font-family:_\\_ Sans\\_';<font-size:10pt;<font-weight:400;<font-style:normal;\\_>\\n"
211         "<p_style=\\_<margin-top:0px;<margin-bottom:0px;<margin-left:0px;<margin-right:0px;<-qt-block-indent:0;<text-indent
                :0px;\\_><span_style=\\_<font-weight:600;\\_>Pulse_Width_(ms)</span></p></body></html>", None, QtGui.
                QApplication.UnicodeUTF8))
212         self.Pulse_width.setToolTip(QtGui.QApplication.translate("Form", "Pulse_width_of_input_pulse\\n"
213         "in_milliseconds_(ms)", None, QtGui.QApplication.UnicodeUTF8))
214         self.Pulse_width.setText(QtGui.QApplication.translate("Form", "0.0001667", None, QtGui.QApplication.
                UnicodeUTF8))
215         self.VPM.setToolTip(QtGui.QApplication.translate("Form", "Plots_Voltage_Propagation_Model_(VPM)_Pulse_
                Shape", None, QtGui.QApplication.UnicodeUTF8))
216         self.VPM.setText(QtGui.QApplication.translate("Form", "Plot_&VPM", None, QtGui.QApplication.UnicodeUTF8))
217         self.plotWidget.setToolTip(QtGui.QApplication.translate("Form", "matplotlib_plot_area", None, QtGui.
                QApplication.UnicodeUTF8))
218         self.label_11.setText(QtGui.QApplication.translate("Form", "m", None, QtGui.QApplication.UnicodeUTF8))
219         self.Close.setToolTip(QtGui.QApplication.translate("Form", "Close_the_application", None, QtGui.
                QApplication.UnicodeUTF8))
220         self.Close.setText(QtGui.QApplication.translate("Form", "&Close", None, QtGui.QApplication.UnicodeUTF8))
221         self.label.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//EN
                \\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
222         "<html><head><meta_name=\\_ qrichtext\\_<content=\\_1\\_</><style_type=\\_ text/css\\_>\\n"
223         "p_{_white-space:_pre-wrap;_}\\n"
224         "</style></head><body_style=\\_<font-family:_\\_ Sans\\_';<font-size:10pt;<font-weight:400;<font-style:normal;\\_>\\n"
225         "<p_style=\\_<margin-top:0px;<margin-bottom:0px;<margin-left:0px;<margin-right:0px;<-qt-block-indent:0;<text-indent
                :0px;\\_><span_style=\\_<font-weight:600;\\_>R_(ohm)</span></p></body></html>", None, QtGui.QApplication.
                UnicodeUTF8))
226         self.label_7.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
                EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
227         "<html><head><meta_name=\\_ qrichtext\\_<content=\\_1\\_</><style_type=\\_ text/css\\_>\\n"
228         "p_{_white-space:_pre-wrap;_}\\n"
229         "</style></head><body_style=\\_<font-family:_\\_ Sans\\_';<font-size:10pt;<font-weight:400;<font-style:normal;\\_>\\n"
230         "<p_style=\\_<margin-top:0px;<margin-bottom:0px;<margin-left:0px;<margin-right:0px;<-qt-block-indent:0;<text-indent
                :0px;\\_><span_style=\\_<font-weight:600;\\_>L_(H)</span></p></body></html>", None, QtGui.QApplication.
                UnicodeUTF8))
231         self.label_3.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
                EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
232         "<html><head><meta_name=\\_ qrichtext\\_<content=\\_1\\_</><style_type=\\_ text/css\\_>\\n"
233         "p_{_white-space:_pre-wrap;_}\\n"
234         "</style></head><body_style=\\_<font-family:_\\_ Sans\\_';<font-size:10pt;<font-weight:400;<font-style:normal;\\_>\\n"
235         "<p_style=\\_<margin-top:0px;<margin-bottom:0px;<margin-left:0px;<margin-right:0px;<-qt-block-indent:0;<text-indent
                :0px;\\_><span_style=\\_<font-weight:600;\\_>C_(C)</span></p></body></html>", None, QtGui.QApplication.
                UnicodeUTF8))
236         self.label_4.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
                EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"

```

```

237 "<html><head><meta_name=\\" qrichtext\\"_content=\\"1\\"/><style_type=\\" text/css\\">\n"
238 "p,li{white-space:pre-wrap;}\n"
239 "</style></head><body_style=\\"_font-family:\\" Sans\\"';_font-size:10pt;_font-weight:400;_font-style:normal;\\>\n"
240 "<p_style=\\"_margin-top:0px;_margin-bottom:0px;_margin-left:0px;_margin-right:0px;_qt-block-indent:0;_text-indent:0px;\\><span_style=\\"_font-weight:600;\\>G(mho)</span></p></body></html>", None, QtGui.QApplication.UnicodeUTF8))
241 self.label_5.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC\\" --//W3C//DTD_HTML_4.0//EN\\"_\\\" http://www.w3.org/TR/REC-html40/strict.dtd\\>\n"
242 "<html><head><meta_name=\\" qrichtext\\"_content=\\"1\\"/><style_type=\\" text/css\\">\n"
243 "p,li{white-space:pre-wrap;}\n"
244 "</style></head><body_style=\\"_font-family:\\" Sans\\"';_font-size:10pt;_font-weight:400;_font-style:normal;\\>\n"
245 "<p_align=\\" justify\\"_style=\\"_margin-top:0px;_margin-bottom:0px;_margin-left:0px;_margin-right:0px;_qt-block-indent:0;_text-indent:0px;\\><span_style=\\"_font-weight:600;\\>Averagetap</span></p>\n"
246 "<p_align=\\" justify\\"_style=\\"_margin-top:0px;_margin-bottom:0px;_margin-left:0px;_margin-right:0px;_qt-block-indent:0;_text-indent:0px;\\><span_style=\\"_font-weight:600;\\>distance(m)</span></p></body></html>", None, QtGui.QApplication.UnicodeUTF8))
247 self.label_6.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC\\" --//W3C//DTD_HTML_4.0//EN\\"_\\\" http://www.w3.org/TR/REC-html40/strict.dtd\\>\n"
248 "<html><head><meta_name=\\" qrichtext\\"_content=\\"1\\"/><style_type=\\" text/css\\">\n"
249 "p,li{white-space:pre-wrap;}\n"
250 "</style></head><body_style=\\"_font-family:\\" Sans\\"';_font-size:10pt;_font-weight:400;_font-style:normal;\\>\n"
251 "<p_align=\\" justify\\"_style=\\"_margin-top:0px;_margin-bottom:0px;_margin-left:0px;_margin-right:0px;_qt-block-indent:0;_text-indent:0px;\\><span_style=\\"_font-weight:600;\\>Lengthof</span></p>\n"
252 "<p_align=\\" justify\\"_style=\\"_margin-top:0px;_margin-bottom:0px;_margin-left:0px;_margin-right:0px;_qt-block-indent:0;_text-indent:0px;\\><span_style=\\"_font-weight:600;\\>txline(m)</span></p></body></html>", None, QtGui.QApplication.UnicodeUTF8))
253 self.R.setToolTip(QtGui.QApplication.translate("Form", "CharacteristicResistanceofthetransmissionline", None, QtGui.QApplication.UnicodeUTF8))
254 self.R.setText(QtGui.QApplication.translate("Form", "0.35", None, QtGui.QApplication.UnicodeUTF8))
255 self.L.setToolTip(QtGui.QApplication.translate("Form", "CharacteristicInductanceof\\n"
256 "thetransmissionline", None, QtGui.QApplication.UnicodeUTF8))
257 self.L.setText(QtGui.QApplication.translate("Form", "0.60e-6", None, QtGui.QApplication.UnicodeUTF8))
258 self.C.setToolTip(QtGui.QApplication.translate("Form", "CharacteristicCapacitanceof\\n"
259 "thetransmissionline", None, QtGui.QApplication.UnicodeUTF8))
260 self.C.setText(QtGui.QApplication.translate("Form", "60.0e-9", None, QtGui.QApplication.UnicodeUTF8))
261 self.G.setToolTip(QtGui.QApplication.translate("Form", "CharacteristicConductanceof\\n"
262 "thetransmissionline", None, QtGui.QApplication.UnicodeUTF8))
263 self.G.setText(QtGui.QApplication.translate("Form", "0.19e-3", None, QtGui.QApplication.UnicodeUTF8))
264 self.N.setToolTip(QtGui.QApplication.translate("Form", "Averagedistancebetweenthetransission\\n"
265 "linejunction/nodepoints", None, QtGui.QApplication.UnicodeUTF8))
266 self.N.setText(QtGui.QApplication.translate("Form", "5", None, QtGui.QApplication.UnicodeUTF8))
267 self.D.setToolTip(QtGui.QApplication.translate("Form", "Totallengthofthe\\n"
268 "transmissionline", None, QtGui.QApplication.UnicodeUTF8))
269 self.D.setText(QtGui.QApplication.translate("Form", "100", None, QtGui.QApplication.UnicodeUTF8))
270 self.x_position.setToolTip(QtGui.QApplication.translate("Form", "Pointofinterestonthe\\n"
271 "transmissionline.\\n"
272 "inputshouldbeinmetres(m)\\n"
273 "formsignalfeedpoint", None, QtGui.QApplication.UnicodeUTF8))
274 self.x_position.setText(QtGui.QApplication.translate("Form", "100", None, QtGui.QApplication.UnicodeUTF8))
275 self.label_2.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC\\" --//W3C//DTD_HTML_4.0//EN\\"_\\\" http://www.w3.org/TR/REC-html40/strict.dtd\\>\n"
276 "<html><head><meta_name=\\" qrichtext\\"_content=\\"1\\"/><style_type=\\" text/css\\">\n"
277 "p,li{white-space:pre-wrap;}\n"
278 "</style></head><body_style=\\"_font-family:\\" Sans\\"';_font-size:10pt;_font-weight:400;_font-style:normal;\\>\n"
279 "<p_style=\\"_margin-top:0px;_margin-bottom:0px;_margin-left:0px;_margin-right:0px;_qt-block-indent:0;_text-indent:0px;\\><span_style=\\"_color:#0000ff;\\>R,L,CandGarethecharacteristic</span></p>\n"
280 "<p_style=\\"_margin-top:0px;_margin-bottom:0px;_margin-left:0px;_margin-right:0px;_qt-block-indent:0;_text-indent:0px;\\><span_style=\\"_color:#0000ff;\\>valuesoftheTransmissionline</span></p></body></html>", None, QtGui.QApplication.UnicodeUTF8))

```

```

281         self.label_12.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
            EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
282 "<html><head><meta_name=\\_<rich_text\\_<content=\\_<1\\_</><style_type=\\_<text/css\\_>\\n"
283 "p,li{white-space:pre-wrap;\\_\\n"
284 "</style></head><body_style=\\_<font-family:Sans\\_<;font-size:10pt;font-weight:400;font-style:normal;\\_>\\n"
285 "<p_style=\\_<margin-top:0px;margin-bottom:0px;margin-left:0px;margin-right:0px;qt-block-indent:0;text-indent
    :0px;\\_><span_style=\\_<color:#0000ff;\\_><Position_of_Interest</span></p>\\n"
286 "<p_style=\\_<margin-top:0px;margin-bottom:0px;margin-left:0px;margin-right:0px;qt-block-indent:0;text-indent
    :0px;\\_><span_style=\\_<color:#0000ff;\\_><(<Distance_from_signal_source</span></p>\\n"
287 "<p_style=\\_<margin-top:0px;margin-bottom:0px;margin-left:0px;margin-right:0px;qt-block-indent:0;text-indent
    :0px;\\_><span_style=\\_<color:#0000ff;\\_><that_pulse_is_viewed)</span></p></body></html>", None, QtGui.
    QApplication.UnicodeUTF8))
288         self.label_10.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
            EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
289 "<html><head><meta_name=\\_<rich_text\\_<content=\\_<1\\_</><style_type=\\_<text/css\\_>\\n"
290 "p,li{white-space:pre-wrap;\\_\\n"
291 "</style></head><body_style=\\_<font-family:Sans\\_<;font-size:10pt;font-weight:400;font-style:normal;\\_>\\n"
292 "<p_style=\\_<margin-top:0px;margin-bottom:0px;margin-left:0px;margin-right:0px;qt-block-indent:0;text-indent
    :0px;\\_><span_style=\\_<color:#0000ff;\\_><InputPulse_Properties</span></p></body></html>", None, QtGui.
    QApplication.UnicodeUTF8))
293         self.IPM.setToolTip(QtGui.QApplication.translate("Form", "Plot_Current_Propagation_Model(IPM)_pulse_shape
            ", None, QtGui.QApplication.UnicodeUTF8))
294         self.IPM.setText(QtGui.QApplication.translate("Form", "Plot_IPM", None, QtGui.QApplication.UnicodeUTF8))
295         self.label_13.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
            EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
296 "<html><head><meta_name=\\_<rich_text\\_<content=\\_<1\\_</><style_type=\\_<text/css\\_>\\n"
297 "p,li{white-space:pre-wrap;\\_\\n"
298 "</style></head><body_style=\\_<font-family:Sans\\_<;font-size:10pt;font-weight:400;font-style:normal;\\_>\\n"
299 "<p_style=\\_<margin-top:0px;margin-bottom:0px;margin-left:0px;margin-right:0px;qt-block-indent:0;text-indent
    :0px;\\_><span_style=\\_<font-weight:600;\\_><Rr(ohm)</span></p></body></html>", None, QtGui.QApplication.
    UnicodeUTF8))
300         self.label_14.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
            EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
301 "<html><head><meta_name=\\_<rich_text\\_<content=\\_<1\\_</><style_type=\\_<text/css\\_>\\n"
302 "p,li{white-space:pre-wrap;\\_\\n"
303 "</style></head><body_style=\\_<font-family:Sans\\_<;font-size:10pt;font-weight:400;font-style:normal;\\_>\\n"
304 "<p_style=\\_<margin-top:0px;margin-bottom:0px;margin-left:0px;margin-right:0px;qt-block-indent:0;text-indent
    :0px;\\_><span_style=\\_<font-weight:600;\\_><Lr(H)</span></p></body></html>", None, QtGui.QApplication.
    UnicodeUTF8))
305         self.label_15.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
            EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
306 "<html><head><meta_name=\\_<rich_text\\_<content=\\_<1\\_</><style_type=\\_<text/css\\_>\\n"
307 "p,li{white-space:pre-wrap;\\_\\n"
308 "</style></head><body_style=\\_<font-family:Sans\\_<;font-size:10pt;font-weight:400;font-style:normal;\\_>\\n"
309 "<p_style=\\_<margin-top:0px;margin-bottom:0px;margin-left:0px;margin-right:0px;qt-block-indent:0;text-indent
    :0px;\\_><span_style=\\_<font-weight:600;\\_><Cr(C)</span></p></body></html>", None, QtGui.QApplication.
    UnicodeUTF8))
310         self.Rr.setToolTip(QtGui.QApplication.translate("Form", "Resistance_of_the_tap_load", None, QtGui.
            QApplication.UnicodeUTF8))
311         self.Rr.setText(QtGui.QApplication.translate("Form", "1e3", None, QtGui.QApplication.UnicodeUTF8))
312         self.Lr.setToolTip(QtGui.QApplication.translate("Form", "Inductance_of_the_tap_load", None, QtGui.
            QApplication.UnicodeUTF8))
313         self.Lr.setText(QtGui.QApplication.translate("Form", "1.3e-3", None, QtGui.QApplication.UnicodeUTF8))
314         self.Cr.setToolTip(QtGui.QApplication.translate("Form", "Capacitance_of_the_tap_load", None, QtGui.
            QApplication.UnicodeUTF8))
315         self.Cr.setText(QtGui.QApplication.translate("Form", "47e-9", None, QtGui.QApplication.UnicodeUTF8))
316         self.label_16.setText(QtGui.QApplication.translate("Form", "<!DOCTYPE_HTML_PUBLIC_<!--W3C//DTD_HTML_4.0//
            EN\\_\\_ http://www.w3.org/TR/REC-html40/strict.dtd\\_>\\n"
317 "<html><head><meta_name=\\_<rich_text\\_<content=\\_<1\\_</><style_type=\\_<text/css\\_>\\n"
318 "p,li{white-space:pre-wrap;\\_\\n"

```



```

319 "</style></head><body_style=\&_font-family:\` Sans\`;_font-size:10pt;_font-weight:400;_font-style:normal;\`>\n"
320 "<p_style=\&_margin-top:0px;_margin-bottom:0px;_margin-left:0px;_margin-right:0px;_qt-block-indent:0;_text-indent
    :0px;\`><span_style=\&_color:#0000ff;\`>Load_Parameters</span></p></body></html>", None, QtGui.QApplication.
    UnicodeUTF8))
321     self.Clear.setToolTip(QtGui.QApplication.translate("Form", "Clears_the_matplotlib_plot_area", None, QtGui.
        QApplication.UnicodeUTF8))
322     self.Clear.setText(QtGui.QApplication.translate("Form", "Clear_Plot_Area", None, QtGui.QApplication.
        UnicodeUTF8))
323
324 from mpl_pyqt4_widget import MPL_Widget

```

A.3 MPL_Widget Listing

```

1  #!/usr/bin/env python
2  from PyQt4.QtCore import *
3  from PyQt4.QtGui import *
4
5  from matplotlib.backends.backend_qt4agg import FigureCanvasQTAff as FigureCanvas
6  from matplotlib.backends.backend_qt4 import NavigationToolbar2QT as NavigationToolbar
7
8  from matplotlib.figure import Figure
9
10 import numpy as N
11
12 class MyMplCanvas(FigureCanvas):
13     def __init__(self, parent=None, width = 10, height = 12, dpi = 100, sharex = None, sharey = None):
14         self.fig = Figure(figsize = (width, height), dpi=dpi, facecolor = '#FFFFFF')
15         self.ax = self.fig.add_subplot(111, sharex = sharex, sharey = sharey)
16         self.fig.subplots_adjust(left=0.1, bottom=0.1, right=0.9, top=0.9)
17         self.xtitle="Time_(ms)"
18         self.ytitle="V/I"
19         self.PlotTitle = "Pulse_Shape_(V/I)"
20         self.grid_status = True
21         self.xaxis_style = 'linear'
22         self.yaxis_style = 'linear'
23         self.format_labels()
24         self.ax.hold(True)
25         FigureCanvas.__init__(self, self.fig)
26         #self.fc = FigureCanvas(self.fig)
27         FigureCanvas.setSizePolicy(self,
28             QSizePolicy.Expanding,
29             QSizePolicy.Expanding)
30         FigureCanvas.updateGeometry(self)
31
32     def format_labels(self):
33         self.ax.set_title(self.PlotTitle)

```

```

34         self.ax.title.set_fontsize(10)
35         self.ax.set_xlabel(self.xtitle, fontsize = 9)
36         self.ax.set_ylabel(self.ytitle, fontsize = 9)
37         labels_x = self.ax.get_xticklabels()
38         labels_y = self.ax.get_yticklabels()
39
40         for xlabel in labels_x:
41             xlabel.set_fontsize(8)
42         for ylabel in labels_y:
43             ylabel.set_fontsize(8)
44             ylabel.set_color('b')
45
46     def sizeHint(self):
47         w, h = self.get_width_height()
48         return QSize(w, h)
49
50     def minimumSizeHint(self):
51         return QSize(10, 10)
52
53     def sizeHint(self):
54         w, h = self.get_width_height()
55         return QSize(w, h)
56
57     def minimumSizeHint(self):
58         return QSize(10, 10)
59
60
61 class MPL_Widget(QWidget):
62     def __init__(self, parent = None):
63         QWidget.__init__(self, parent)
64         self.canvas = MyMplCanvas()
65         self.toolbar = NavigationToolbar(self.canvas, self.canvas)
66         self.vbox = QVBoxLayout()
67         self.vbox.addWidget(self.canvas)
68         self.vbox.addWidget(self.toolbar)
69         self.setLayout(self.vbox)

```

Index

- attenuation coefficient, 13
- boundary conditions, 16
- capacitance, 7
- cascaded blocks, 28
- characteristic equations, 11
- characteristic resistance, 10
- Coaxial, 5
- Conclusion, 55
- Current Propagation Model, 33
- Current Propagation Model Flowchart, 40
- discretized, 28
- Equivalent Circuit, 7
- Euler's Identity, 13
- Fourier transform, 16
- frequency domain, 15
- Frequency Domain Solution, 15
- geometry of the transmission, 6
- GUI, 3
- GUI Python Listing, 79
- Impedance of a Transmission Line With Losses, 8
- Impedance of an Ideal Transmission Line, 7
- infinitesimal, 9, 10
- IPM, 33, 54
- IPM Equation, 34
- KCL, 12
- Kirchoff's Current Law, 12
- Kirchoff's Voltage Law, 11
- KVL, 11
- loss tangent, 6
- Main Program Listing, 59
- Matplotlib, 37
- Model Assumptions, 27
- Model Description, 28
- Motivation, 3
- MPL_Widget Listing, 86
- Multi-access, 23
- Node model, 24
- Nokia, 3
- Numpy, 37
- phase shift coefficient, 13
- Phase Velocity, 19
- Power Propagation Model, 36
- power transmission line, 3
- PPM, 36
- propagation constan, 13
- Propagation Constant, 19
- Pulse.ui Listing, 66
- Pulses, 19
- PyQt, 37
- Python, 2, 37
- Qt, 3
- Qt Designer, 37
- Recommendation, 55
- Results Discussion, 53
- Scipy, 37
- series, 28
- Series and Parallel Losses, 10
- series losses, 9
- shunt, 28
- Signal analysis, 15
- Simulation Results, 43
- Simulation tools, 37
- Solution to Differential Equation, 18
- Solution to the Transmission Line, 13
- Spyder, 38
- Step Function Response, 20
- Stripline, 5
- The GUI of the Model program, 42
- transmission line design, 8
- Transmission Line Equations, 10
- Transmission Line Theory, 5
- Two-wire, 5
- Voltage Propagation Model, 29
- Voltage Propagation Model Flowchar, 38
- VPM, 29, 53
- VPM Equation, 30