GAS KICK MECHANISTIC MODEL

A Thesis

by

RAHEEL ZUBAIRY

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	A. Rashid Hasan
Committee Members,	Jerome J. Schubert
	Debbie Thomas
Head of Department,	A. Daniel Hill

May 2014

Major Subject: Petroleum Engineering

ABSTRACT

Gas kicks occur during drilling when the formation pressure is greater than the wellbore pressure causing influx of gas into the wellbore. Uncontrolled gas kicks could result in blowout of the rig causing major financial loss and possible injury or loss of lives.

The influx of gas into the wellbore and the subsequent migration of gas towards surface affect different parameters related to the control of the wellbore such as wellbore's annulus pressure profile, mud density, pit-gain and temperature profile in the annulus. This research focuses on these changes in these parameters to be able to detect the occurrence of gas kick and the circulation of the gas kick out from the well. In this thesis, we have developed a model that incorporates a mechanistic approach to determine the transition of the flow as the gas migrates to surface. The model is implemented in a simulation program to present the behavior of these parameters.

The simulator initially creates a profile of the mud-circulation in the well-bore using user specified geometry of the well, drilling mud's characteristics and temperature properties of the formation and the drilling mud. The simulator then uses user-specified estimation of the formation pressure and other formation properties at a user-specified depth to calculate the gas-inflow into the wellbore. From this gas flow into the wellbore, the simulation shows changes in the annulus pressure and pit gain at surface as function of time, in addition to the changes in temperature profile and mud properties as gas migrates to surface.

ACKNOWLEDGEMENTS

I would like to express my gratitude to the chair of my graduate advisory committee, Dr. A. Rashid Hasan, for his continuous guidance and support during the course of this research.

I would like to thank Dr. Jerome J. Schubert and Dr. Debbie Thomas, for agreeing to be on my advisory committee and reviewing this work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Gas kick occurs during drilling operations when the formation pressure exceeds the well-bore pressure causing influx of gas from the formation into the wellbore. The dominant reason for gas kicks is insufficient mud weight which results in formation pressure exceeding the well-bore pressure. On the other hand too much overpressuring the wellbore using heavy mud-weight is not a viable solution as it can induce fractures into the formation causing loss of circulation and formation damage. Uncontrolled gas kicks can result in blowouts causing significant financial losses, damage to the environment, and possibly injury or loss of life. Thus, early detection of a gas kick is very important as it can help the operators to respond quickly with appropriate well control measures in an effective manner.

Currently, the main indicators of gas kicks are increase in flow rate out of the annulus, increase in pit gain volume, abrupt increase in drilling rate, well flowing with pumps off, and changes in the circulating pump pressure. These indicators are taken into account as a whole by the operator and the operator's experience to decide whether to initiate well control measures where slight misjudgment could result in a blowout. Taking well control measures, however, also result in drilling delays therefore increased cost. Thus, the operator also needs to balance the decision to take preventive measures to avoid a possible blowout against cost incurred in taking such measures. There is a need to understand the effects of gas kick on the different parameters of mud flow, which could provide better well control response to gas kicks.

Most of the existing work on the topic focuses on circulating the gas kick out after it has been detected and use empirical two-phase methods to calculate the two phase behavior. This research provides a mechanistic model to estimates changes in pressure, temperature and fluid property in the well-bore when the gas kick occurs so as to better understand the process and develop better inidicators for an early detection of a gas kick. The model is being simulated using Excel and Visual Basic for Applications (VBA). There are specific input Excel sheets through which the program creates profiles, simulated data and charts in specific output sheets. The calculations are performed in VBA to create a robust program and user-friendly environment.

The simulator begins with calculations of pressure and temperature profile in the tubing and the annulus during mud-circulation. The user specifies the wellbore geometry which includes casing till a certain depth, length of drill collars, and measured values of bit nozzles. The simulator uses specified drilling mud's flow rate, density and viscometer readings which are part of a mud engineer's report. For temperature profile calculations, the simulator uses user-specified thermal conductivity and specific heat of the drilling mud and formation, in addition to a surface temperature, mud in-let temperature and the geothermal gradient.

The mud-circulation profile for the single phase drilling mud uses the Power law model to calculate mud viscosity in the tubing and the annulus. The model provides for the change in viscosity with the changing flow area as the mud flows down the tubing and up the annulus. The pressure profile for the mud-circulation in the tubing and annulus incorporates the frictional and the hydrostatic pressure gradient. The profile shows the pressure increase in the tubing as mud flows down, followed by the pressure loss in the bit and then the pressure drop in the annulus as mud flows up the annulus.

Currently, the existing models on gas kick simulation use constant temperature or incremented temperature in the wellbore. This research provides for a temperature model incorporating the work of Hasan et al. (1996) for temperature calculation during mud-circulation in the wellbore. The model accounts for the geothermal temperature surrounding the wellbore and the temperature difference between the tubing and annulus. The model provides for calculation for annulus and tubing temperature by doing an energy balance for the fluid.

After the mud circulation profiles are created, the user specifies formation pressure and properties. Based on the difference in the pressure between the reservoir and annulus pressure, at a specified depth of entry, a gas kick is instigated into the annulus. The gas kick alters the properties of the mud flow in the annulus by occupying a certain area of the annulus. This affects the pressure drop and fluid properties in the annulus. As the gas rises towards surface, the gas expands due to the lower pressure. This expansion further affects the flow in the annulus.

The two-phase pressure loss in the annulus is modeled using the mechanistic approach presented by Hasan and Kabir (2002). Their approach specifies the two phase flow into a particular flow regime depending on the flow of gas in the liquid flow stream. Through this approach, one can estimate the gas void fraction which is the fraction of volume occupied by gas in the stream. The gas void fraction strongly influences the mixture density and mixture viscosity, which in turn impacts the pressure drop in the annulus. This research assumes no solubility of gas into the liquid stream thus being applicable for water based mud as oil based mud shows tendency of gas solubility.

The temperature model shows changes in the temperature profile as the gas flow into the annulus. The model takes into account the effect of gas inflow at geothermal temperature and the increased flow in the annulus after the gas entry. The model does not account for heat transfer between the single phase mud and the two phase mixture.

This research shows as a function of time how the annulus pressure and pit gain alter as the gas kick migrates to surface. This is performed by calculating pressure drop at each length increment and taking single phase pressure drop for each increment above as gas migrates up. At each increment, the simulator calculates the time it takes for the gas to travel up. The pit gain is calculated using the gas void fraction at each increment to estimate the volume displaced by the gas. Fig. 1 displays the mud circulation system and the increase in mud pit as gas enters the wellbore.

This research provides for detection of gas kick at specific reservoir pressure primarily through changes in the annulus pressure profile, fluid properties and pit gain. The research also provides for annulus pressure estimation when considering failure of casing and cement down hole after a gas kick occurrence. In addition, this simulator could be used for modeling driller's method for well control. The driller's method is implemented after the well has been shut-in due to gas kick occurrence. This method circulates the gas kick out of the wellbore using the same mud weight while keeping the bottom hole constant.

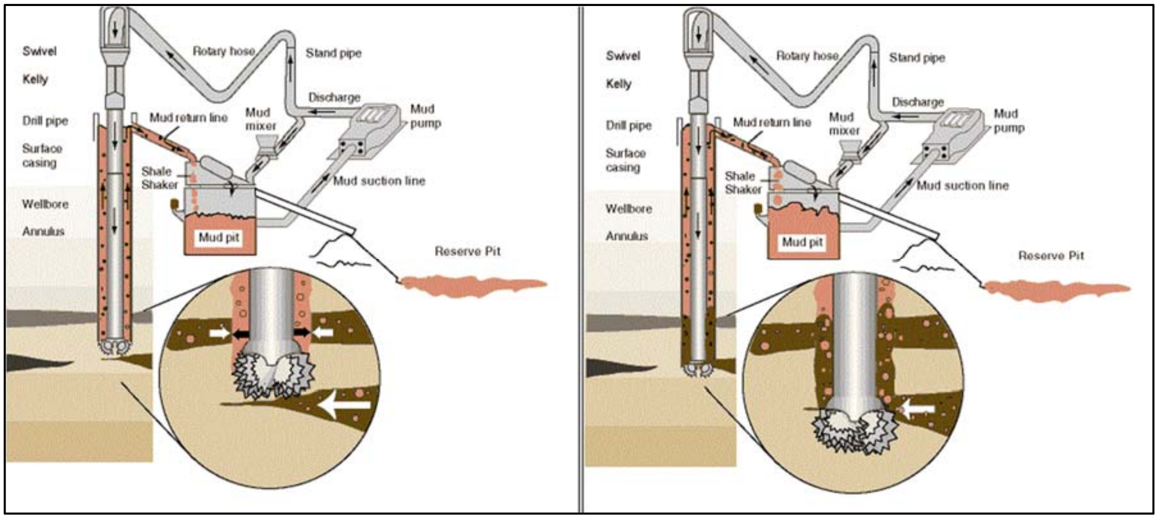

Fig.1 - Effect of gas kick on mud pit and the mud circulation system (United States Department of Labor, 2003)

The goal of this research is to calculate changes in pressure and mud flow in the annulus after a gas kick occurs. The temperature profile provides for more precise temperature estimation at each depth increment. The research shows as a function of time how pressure profile in the annulus and pit gain increases. As all of the parameters of the two phase flow are strongly dependent on the gas void fraction in the annulus, the mechanistic approach provides for a more scientific approach as compared to empirical methods.

LITERATURE REVIEW

The literature review for this research covered the existing models which are primarily for circulating the gas kick out of the well after the gas kick is detected with gas kick volume estimated by the pit gain before the shut-in. Other research includes the gas rise behavior in the annulus and effects on bottom-hole pressure when gas kick occurs while keeping the surface annulus pressure constant.

Initial work on the subject includes work by Leblanc and Lewis (1968) and Rader et. al (1975). Leblanc and Lewis (1968) presented a model to predict the annular back pressure as the gas kick migrates up the surface. They treat the gas in the annulus as a continuous slug occupying the whole cross-sectional area and taking gas velocity same as the mud. They show the annular back pressure curve as new mud is circulated and how it is affected by changing different parameter. Rader et. al (1975) presented a study of the behavior of gas bubble as gas rises in the annulus during a gas kick. The work focused on different factors affecting the bubble rise velocity such as annular geometry, gas and liquid densities, rate of gas expansion, liquid velocity and bubble length.

Choe (1995) provides through well control models in his thesis covering different methods for mud circulation and gas flow in the annulus. The models are based on circulating the gas kick out after the well has been shut-in. The calculations use shut-in drill pipe pressure, shut-in casing pressure and pit-gain to estimate the gas volume in the wellbore. For the two-phase method presented in circulating the gas kick out, Choe

(1995) uses the initial two-phase model presented by Hasan and Kabir (1986) after an intensive literature survey. There have been critical improvements made to the model which is employed for the model presented in this research.

Omosebi et. al (2012) present predictions for annular pressure during well control using driller's method and engineer's method. They consider two cases: when the gas kick flows as a continuous slug and when gas kick is mixed with the mud. Starrett et. al (1990) model provides predictions of bottom hole pressure while keeping surface pressure as atmospheric when gas kick occurs. The model incorporates formation properties and calculates gas influx as transient radial gas flow for an infinite reservoir. The model uses Aziz et. al (1972) two phase correlation to determine liquid holdup and two-phase frictional pressure drop.

Avelar et. al (2009) presents a deepwater gas kick simulation which model the gas kick and well control problem in three stages: drilling fluid prior to gas entrance, the formation gas entrance into the well (Darcy's law radial flow), and application of driller's well control method until all gas is expelled from the well. The simulation was based on two cases: constant bottom hole pressure and a bottom hole pressure gauge reading used as a boundary condition. Santos (1991) presented a model of gas kicks for horizontal well which breaks the well into three regions: a single phase region where the displacing drilling fluid flows behind the gas zone under steady-state conditions, a two phase region where the two-phase mixture flows under unsteady state conditions and a single phase region where the drilling fluid flows ahead of the gas kick zone, which is accelerated by gas expansion as it moves toward the surface.

Nunes et. al (2002) provides a model for deep water wells considering the water depth in calculating pressure profiles. Nunes et. al (2002) calculates the gas velocity considering the gas influx as a slug flow pattern, where the top of gas kick is considered as a Taylor bubble and tail of gas kick as bubble flow.

Nickens (1987) provided a model to stress the importance of realistic assumptions and to illustrate the flexibility of a transient well-control model. The model provides transient nature of different parameters such as casing shoe pressure surface pressure and pit gain as the kick is circulated out through different well control measures. In the model, single-phase friction uses power-law model of mud viscosity to calculate friction factor, while the two phase friction pressure gradient is calculated using the largely empirical Beggs and Brill correlation (1973). Nunes et. al (2002), Avelar et. al (2009) and Santos(1991) also use the Beggs and Brill correlation (1973) for the friction pressure loss in the two phase region.

MUD CIRCULATION MODEL

This simulator creates an initial profile of the wellbore during mud circulation before the gas kick. During drilling, mud is circulated down into the drill pipe, which goes through the bit nozzles and back up the annulus to the surface as shown in Fig. 2. At the surface, the mud flow exits the annulus through a mud return line and enters the mud pit through shakers and other filters. From the mud pit, the mud is circulated back into the wellbore. The mud goes through the suction line and using mud pumps to flow through the stand pipe and and into the drill pipe.

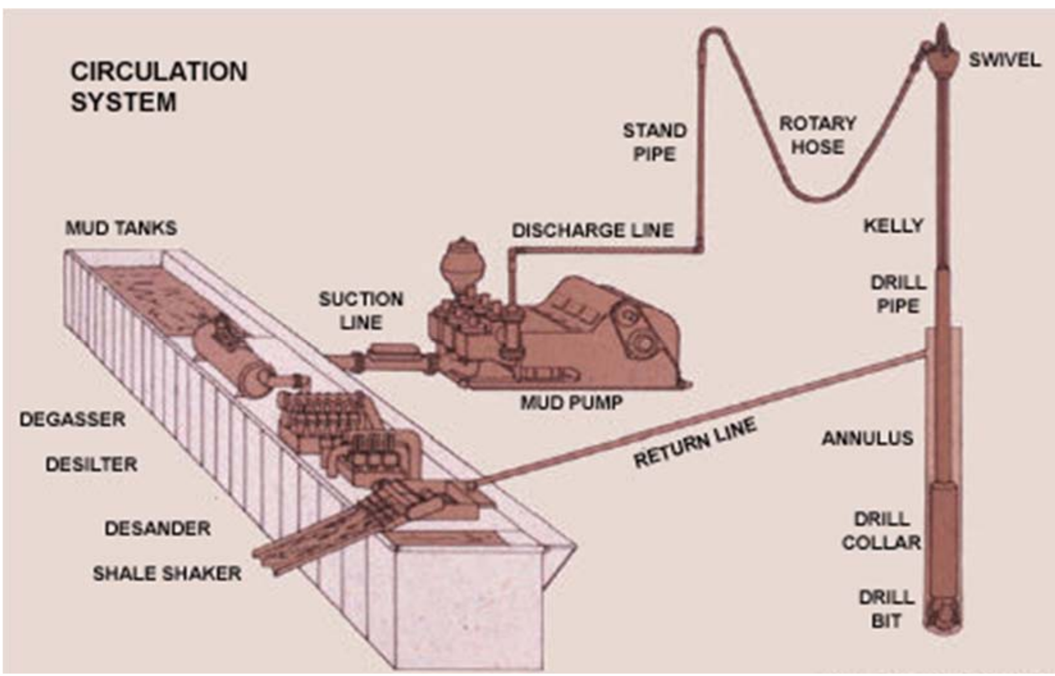

Fig. 2 – Mud circulation system (United States Department of Labor, 2003)

The user-specified inputs for the well geometry includes single casing till certain depth and drill collars as above the bit. The simulator accounts for the changes in tubing and the annulus diameter. If the user specifies a certain length outside of the length increments, the simulator breaks up the increment to account for the change in geometry. The change in diameter effects estimation of mud viscosity and subsequently the pressure drop calculation. Table 1 shows the user specified input for the simulator including well geometry, mud density, mud flow rate, length increment and bit nozzle length. The bit nozzle lengths provide for the pressure drop at the bit.

Table 1 – Input data for wellbore geometry, mud flow and bit nozzle length

INPUT DATA	
Geometry	
Well Depth (ft):	10000
Depth of casing seat (ft):	5000
Inner diameter of casing (in.):	11
Open hole diameter (in.):	9.5
Length of drill-collars (ft):	600
OD drill-collar (in.):	7
ID drill-collar (in.):	2.5
OD drill-pipe (in.):	5
ID drill-pipe (in.):	4.5
Pipe rioughness:	0.00015
Length Increment (ft):	100
Mud flow	
Mud flow rate (gal/min):	200
Mud density (lb/gal):	12
Bit nozzle	
Bit nozzle (32nds in):	12
Bit nozzle 2 (32nds in):	12
Bit nozzle 3 (32nds in):	12

The the drilling fluid's rheological characteristic vary from gelled, Non-Newtonian fluid to a completely Newtonian fluid. Currently in the industry, the three most popular methods to determine rheology of drilling fluids are the Power Law, Herschel-Buckley and Bingham Plastic. This model uses the Power Law method to calculate the viscosity of the drilling fluid in the wellbore as shown in Appendix A.

Table 2 – Input data from mud viscometer reading and temperature characteristics of mud and formation

INPUT DATA	
Mud viscometer reading	
R3:	3
R100:	20
R300:	39
R600:	65
Temperature data	
Inlet mud temperature, °F	70
Mud thermal conductivity, Btu/(ft-°F-hr)	1
Mud specific heat, Btu/(lbm-°F)	0.4
Formation thermal conductivity, Btu/(ft-°F - hr)	1.3
Formation specific heat, Btu/(lbm-°F)	0.2
Formation density, lbm/ft3	165
Surface earth temperature, °F	50
Geothermal gradient, °F/ft	0.0127
Circulation time, hrs	44

The user-specified viscometer readings are common on a drilling mud report. Here the R3 is the viscometer reading at 3 rev/min, R100 is the viscometer reading at 100 rev/min and so forth.

The pressure loss calculation takes into account the hydrostatic and frictional pressure loss. The viscosity of the drilling mud is used in calculating the frictional pressure loss. The pressure drop calculation in tubing adds the hydrostatic flow and subtracts the frictional pressure drop due to downward flow. For pressure calculation in the annulus, the hydrostatic pressure drop and frictional pressure drop are added because of upward flow.

The pressure drop in the bit is dependent on the bit-nozzle size. The simulator assumes that a bit with three nozzles is being used during drilling. The calculations for pressure loss at the bit are shown in Appendix A. The Slow Pump Pressure during circulation is displayed as sum of total frictional pressure loss during circulation of mud and the pressure loss at the bit. This pressure is taken as the initial tubing pressure at surface.

Fig. 3 shows the pressure increase in the drillpipe as the well depth increases with mud following into the drillpipe. This is followed by pressure loss at the bit (given by the divergence between the tubing and the annulus pressure at the bottom of the well). The figure illustrates the pressure drop in the annulus as circulating mud flow up to the surface.

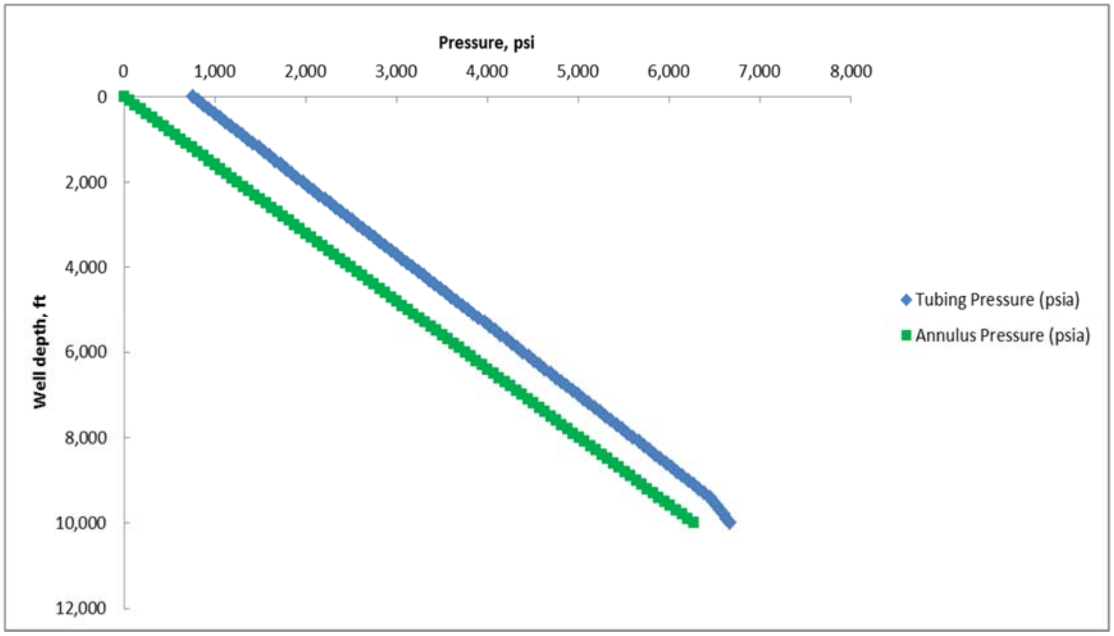

Fig. 3 – Pressure profile of single phase mud through tubing and up the annulus

All the models from literature review for this research assume constant temperature or user-specified temperature range for the wellbore. The simulator incorporates temperature profile for mud-circulation presented by Hasan et al. (1996). Their work arrives at tubing and annulus temperature by doing an energy balance for the fluid. The model accounts for heat transfer from the formation into the annulus from which it gains heat, and the tubing fluid temperature to which it loses heat as shown in Fig. 4. The model assumes constant mud-tank temperature and constant hole and tubing diameter. The user-specified input used includes primarily the specific heat and thermal conductivity of mud and formation, the surface temperature, the inlet mud temperature and a circulation time as show in Table 2. The model provides for a more methodical approach in estimating temperature in the well-bore.

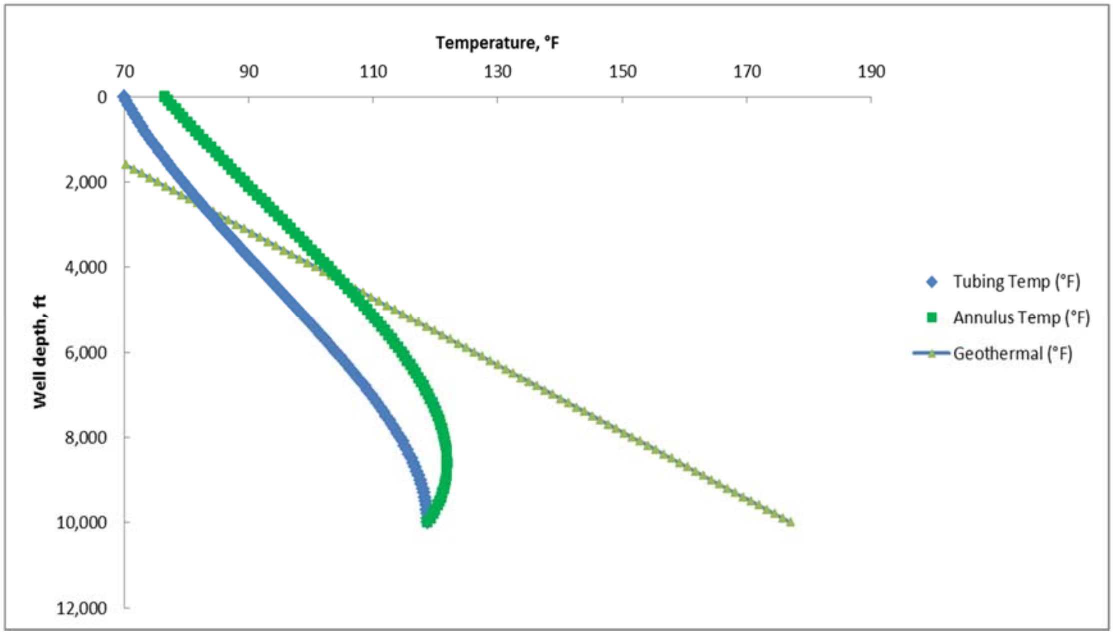

Fig. 4 – Temperature profile of circulating mud while drilling

GAS KICK INFLUX

After a pressure and temperature model of the mud circulation is created, the model intakes a user-specified depth for the entry of gas into the annulus. The gas influx is calculated use Darcy flow by user specified formation properties and using the pressure difference between formation pressure and annulus pressure to calculate the gas inflow. Fig. 5 shows the influx from the reservoir into the annulus, as drilling mud flows from the tubing into the annulus through the bit.

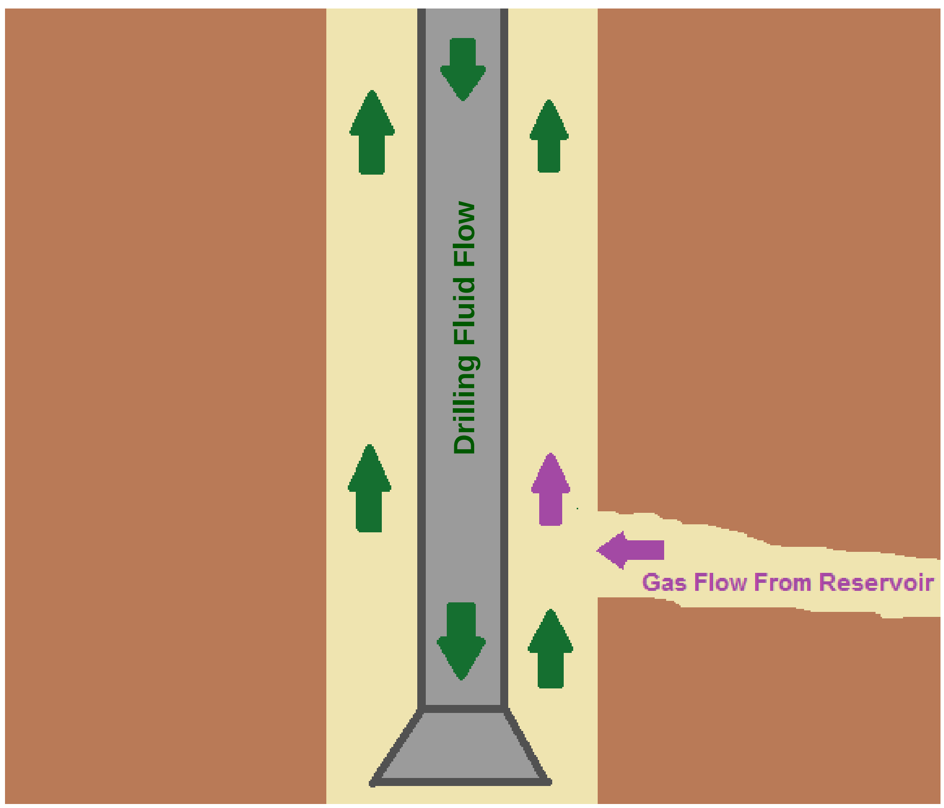

Fig. 5 - Gas kick influx into the wellbore

The gas properties to estimate the Z deviation factor and viscosity are shown in Appendix A. The temperature for the gas influx is the geo-thermal gradient at the depth of the entry. The reservoir height for this simulation is user-specified, however would be best estimated as difference between the depth of gas entry and well depth. This would account for hole drilled into the gas reservoir.

Table 3 – Input data for gas influx into the wellbore

INPUT DATA	
Depth of entry (ft)	9990
Reservoir pressure (psi)	6600
Formation permeability (md)	1
Reservoir height (ft)	10
re (ft)	1000
skin	0
Gas gravity	0.7

TWO-PHASE MODELING

When gas enters the wellbore, there is an initial increase in annulus pressure at well-head before the bottom hole pressure drops. This research focues on the increases in annulus pressure, keeping bottom hole pressure constant. From the depth of entry of gas into the annulus till surface, the model calculates the two phase fluid properties and the pressure drop. The calculations are made at each length increment till gas reaches surface. The simulator assumes constant bottom-hole pressure and single-phase flow for well depth below the gas influx.

The model calculates the pressure drop in the annulus of the two-phase using the mechanistic method provided by Hasan and Kabir (2002). The method is based on finding the velocity of both phases and the area occupied by each phase. The method accounts for slip velocity between gas and liquid flow. The calculations use the superficial velocity of gas and liquid and their respective densities. The superficial velocity of any phase is the volumetric flow rate of that phase over the total cross-sectional area. The actual velocity of the phases would be greater than the superficial velocity, as gas and liquid compete for the available flow area. The area occupied by gas as fraction of the total area is called the gas void fraction, and the area occupied by the liquid as a fraction of the total area is known as liquid holdup.

The estimation of precise gas void fraction is critical in determining the two-phase flow characteristics. The mixture density and viscosity estimation are strongly dependent on gas void fraction, which determine the magnitude of pressure drop. The current methods employed to calculate the two-phase flow for gas kick simulator are largely empirical. Empirical approaches can potentially give inaccurate results when applied to different circumstances from the data they were derived from.

This mechanistic model presented by Hasan and Kabir (2002) separates the flow characteristic into five flow regimes for estimation of gas void fraction. The five flow regimes are: bubbly, slug, churn, dispersed bubbly and annular. These flow regimes are dependent on superficial velocity of gas, superficial velocity of liquid, and the two-phase's characteristics.

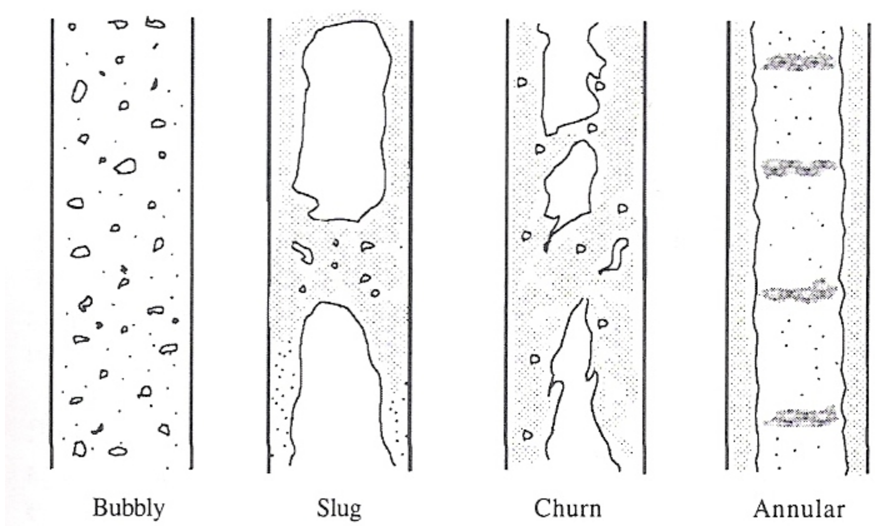

Fig. 6 - Separation of flow regimes (Hasan and Kabir, 2002)

Fig. 4 shows the separation of the flow regimes for increasing superficial velocity of liquid and gas at standard conditions. The model by the Hasan and Kabir (2002) provides how the gas void fraction is calculated at each flow-regime building on work by Harmathy (1960), Wallis (1969) and Zuber and Findlay (1965). Their mechanistic model for two-phase flow has shown good agreement with majority of the empirical models and provides for a more scientific approach compared to the existing two-phase models.

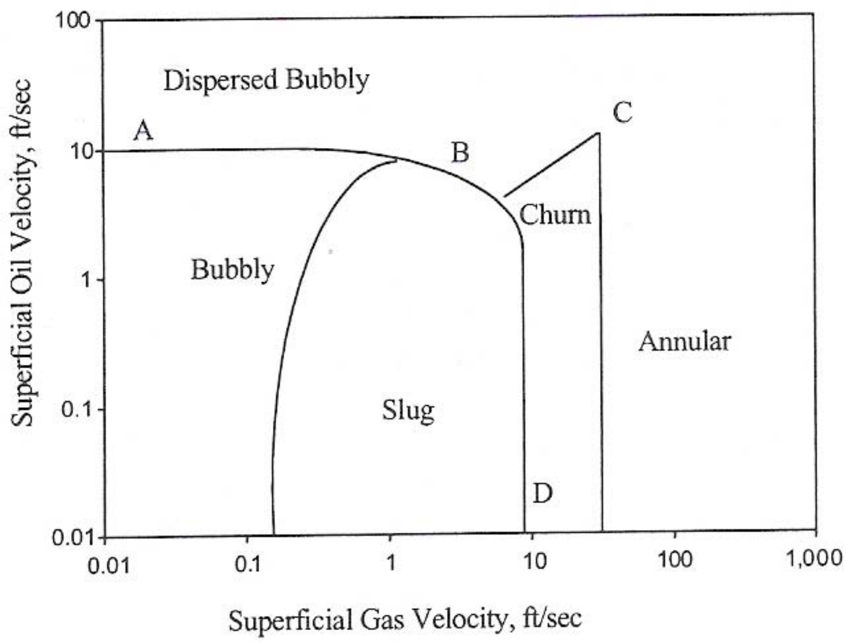

Fig. 7 - Flow regime map (Hasan and Kabir, 2002)

The two-phase pressure drop takes into account the two-phase hydrostatic pressure drop and frictional pressure drop. The hydrostatic pressure drop accounting for majority of the pressure drop depends significantly on the mixture density. The mixture density accounts for gas in the two phase flow using gas-void fraction. The method used to determine the flow regime, and calculations for the two-phase density and pressure drop are presented in Appendix A.

The comparison in the annulus pressure before and after the gas kick occurs as the gas kick rises to surfaces is shown in Fig. 5. Note that the annulus pressure after gas kick increases as the gas migrates up the annulus. This is due to decrease in pressure drop in the annulus as low-density gas infuses into the circulating mud causing the fluid density in the annulus to drop. This reduces the hydrostatic pressure drop in the two-phase flow compared to single phase circulating mud due to lower fluid density. As the gas kick migrates upwards, the two-phase fluid density decreases further due to gas expansion as result of lower pressures on the mixture when approaching surface. This further reduces the hydrostatic pressure drop in the two-phase flow, thus increasing the overall pressure compared to just single-phase mud flow in the annulus.

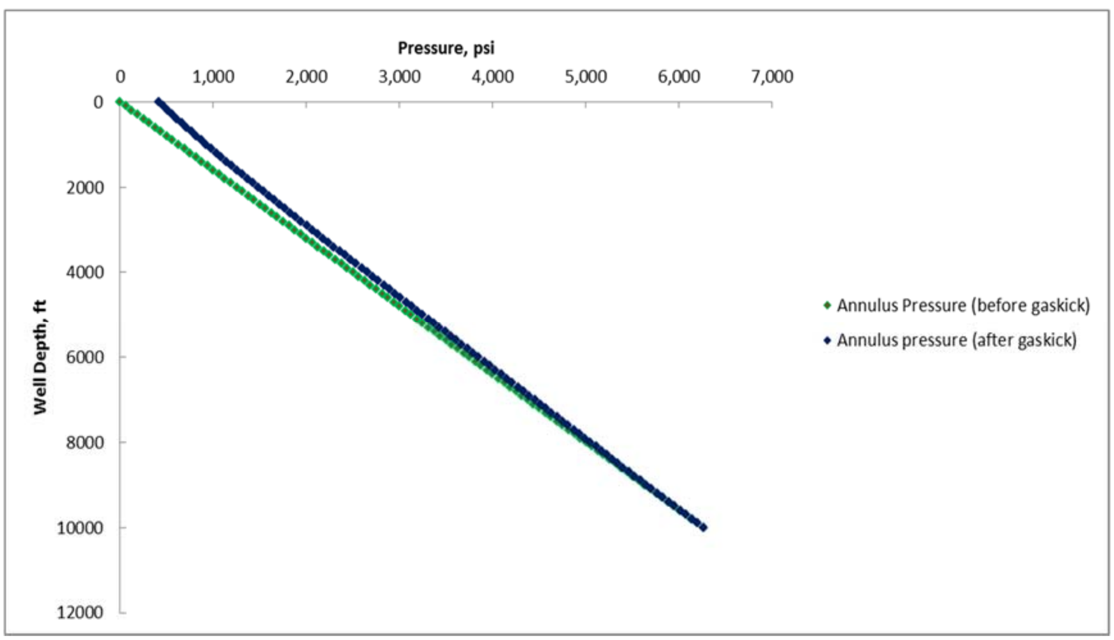

Fig. 8 - Difference in pressure profile in the annulus before and after gas kick

The effects of gas migration on temperature in the annulus are shown in Fig. 6 as the gas rises to surface. For the changes in the temperature, the model accounts for the change in mass flow and specific heat of the mixture. The change in mass flow uses mixture density flowing through the annulus area at mixture velocity. The specific heat for the two phase flow is computed by finding an average using the liquid and gas densities and their respective flow rates. The model also incorporates the temperature increase/decrease from gas influx at geothermal temperature. The total energy at the point of gas entry is calculated by adding the heat energy provided by liquid and gas flow. This provides the temperature difference due to gas influx which is very slight due to the low density of gas. The equations used are presented in Appendix A.

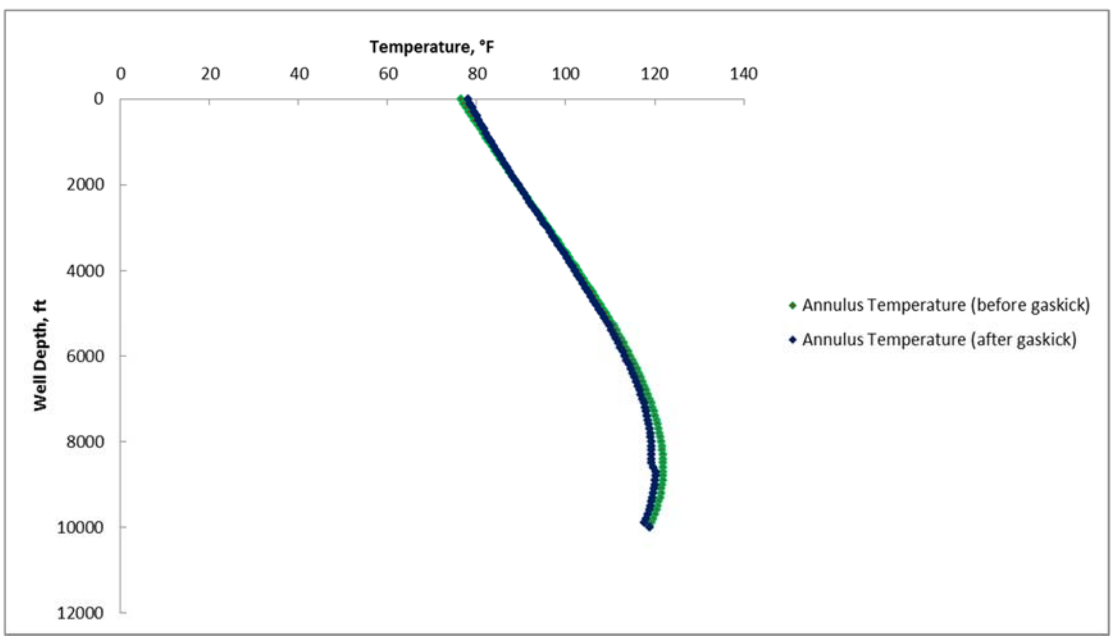

Fig. 9 - Difference in temperature profile in the annulus before and after gas kick

FUNCTION OF TIME

The model provides as function of time, changes in annulus pressure profile and pit gain as the gas kick migrates to surface. This is accomplished by taking time steps at each increment as the gas travels up. The time step is calculated using the actual gas velocity at each increment. The actual gas velocity is calculated by dividing the superficial gas velocity by the gas void fraction, since the superficial gas velocity is flow rate over the entire flow area,

Using these time-steps at each increment, the model is able to display how the annular pressure profile and pit gain changes over time. In addition, this method stimulates when the mud properties and temperature change at a specific depth as the gas migrates up to surface.

The annulus pressure profile calculations are performed by using time-steps and the two-phase pressure drop at each increment as gas travels up the annulus. After the increment, the model calculates single-phase pressure drop in the annulus for the length increments till surface. Here again, the bottom-hole pressure and pressure below the depth of gas entry are maintained constant, and the annulus pressure at zero at surface as mud flow from the mud return line into the pit. This loop runs till the gas reaches surface. The annulus pressure at wellhead as a function of time is shown in Fig. 7. A similar display is shown in Fig. 8 for the mud pumped during the interval as the annulus pressure at well head rises.

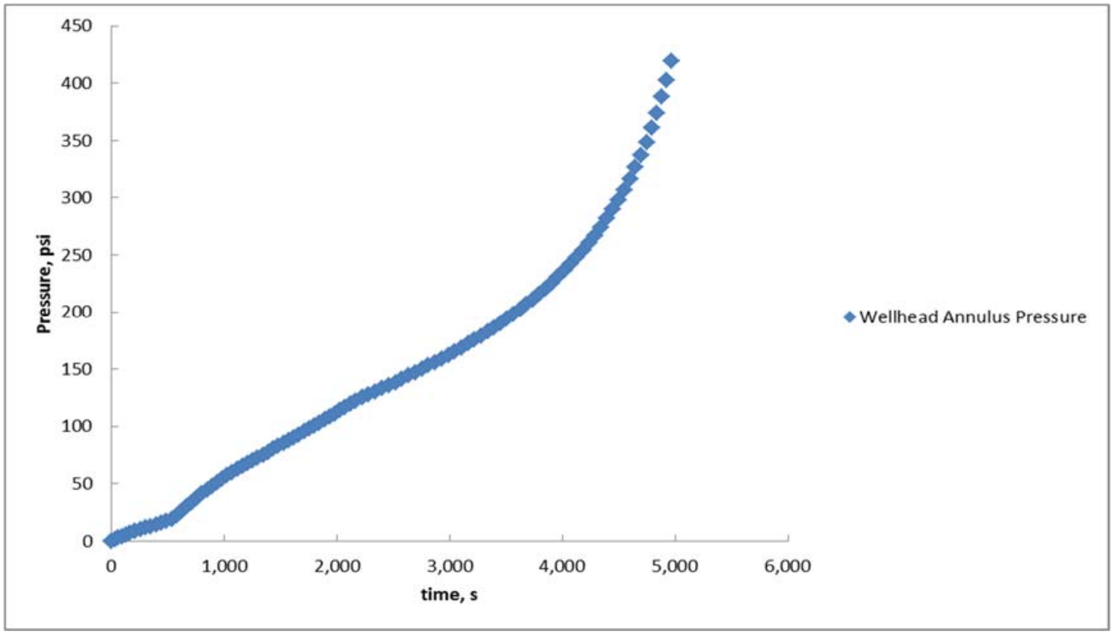

Fig. 10 – Annulus pressure at wellhead against time

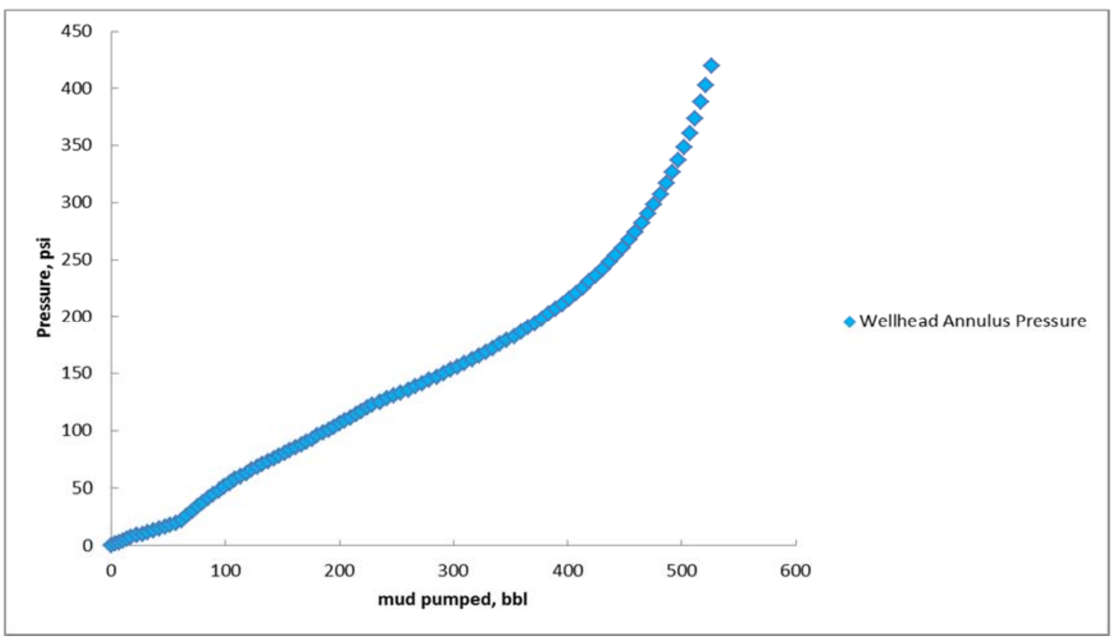

Fig. 11 – Annulus pressure at wellhead against mud pumped

The pit gain is calculated for each time step as volume displaced by gas in the annulus. This is calculated using the gas void fraction calculated at each increment to determine the volume of gas. As the gas rises above, the pit gain volume account for the volume displaced by gas at all the increments below. The pit gain for the simulated well, as a function of time and mud pumped are shown in Fig 9 and Fig. 10. The trend is similar to the annulus pressure as function of time at wellhead.

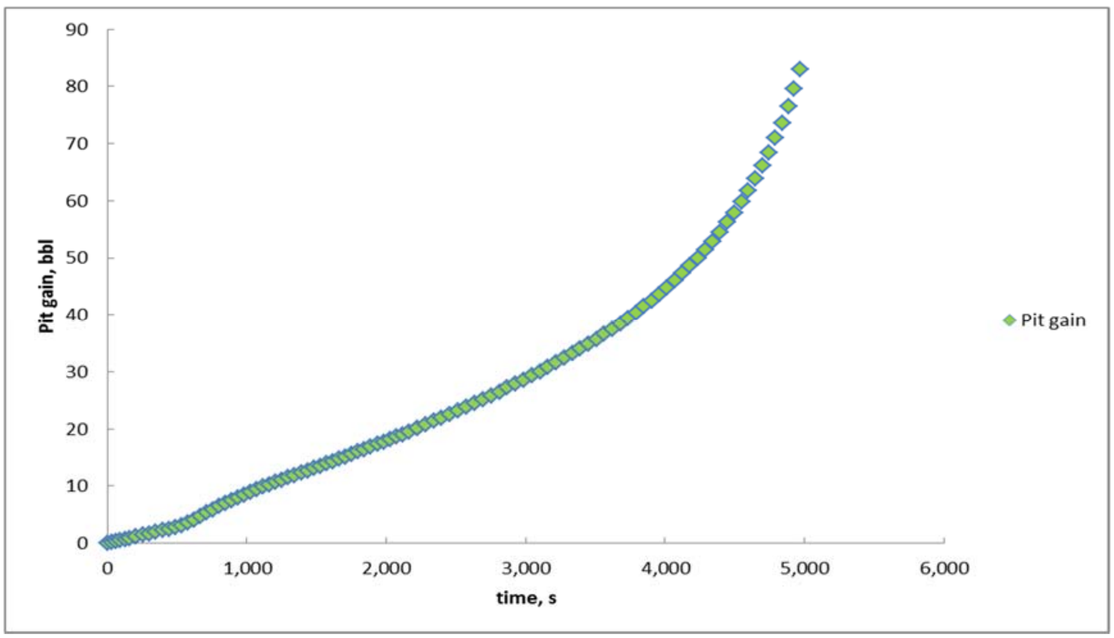

Fig. 12- Pit gain against time

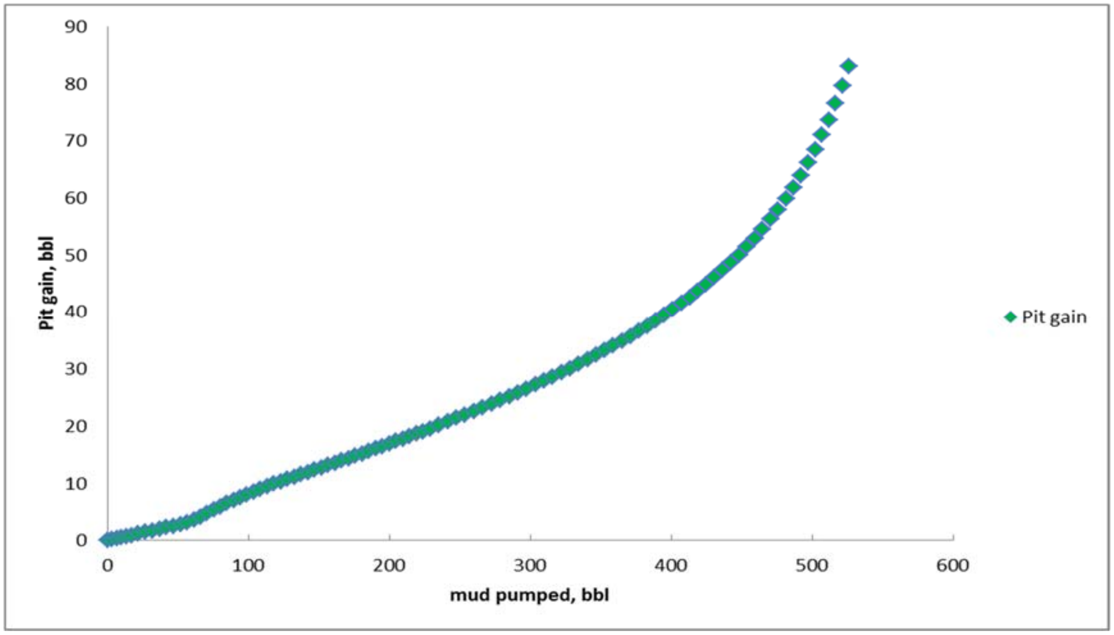

Fig. 13 - Pit gain against mud pumped

WELL CONTROL APPLICATION

Key goal of this model is to assist in gas kick detection and well control measures in circulating gas kick out of the well-bore. The premise of the model is for the user to model the influence of gas kicks on certain wellbore parameters through specific reservoir pressure at certain depth. The model would allow the user to predict the gas flow in into the annulus and its subsequent effect on annulus pressure profile, fluid properties and pit gain. The user would be better able to prepare for well-control measures.

As the technology in pressure sensing and temperature sensing is advancing, these sensors could be used in predicting gas kick through certain changes. This simulation can provide for sensing gas kick through pressure and temperature sensors to detect gas kick and its magnitude. The simulator shows a pattern for how temperature and pressure alter as gas kick migrates up the surface.

Currently, most of the existing gas kick models are based on circulating the gas kick out of the wellbore after it has been detected. After a gas kick is detected, the immediate procedure is to shut-in the well using blow out preventer (BOP). After the well is shut-in, the operator calculates the mud weight needed to get the wellbore pressure higher than the reservoir pressure. The gas is then circulated out through a choke line, which is under the blowout preventer. There are two methods in circulating the gas kick out: the driller’s method and the engineer’s method. The driller’s method uses the original mud to circulate the gas kick out of the well before the heavier mud is pumped in while the engineer's method uses the heavier mud to circulate the gas kick out.

The simulator presented here is limited for this purpose, however shows similarity when circulating the gas kick out. The simulator assumes constant bottom hole pressure which is important for circulating the gas kick out using driller's method or engineer's method. However, the simulator assumes constant gas inflow into the wellbore as compared to assuming fixed gas volume in the wellbore. The inititial choke pressure is user-specified for the mechanistic model and then models the increase in pressure at wellhead which would be same as the choke pressure. Fig. 11 shows a comparison between the driller's method presented by Choe (1996) and the mechanistic model.

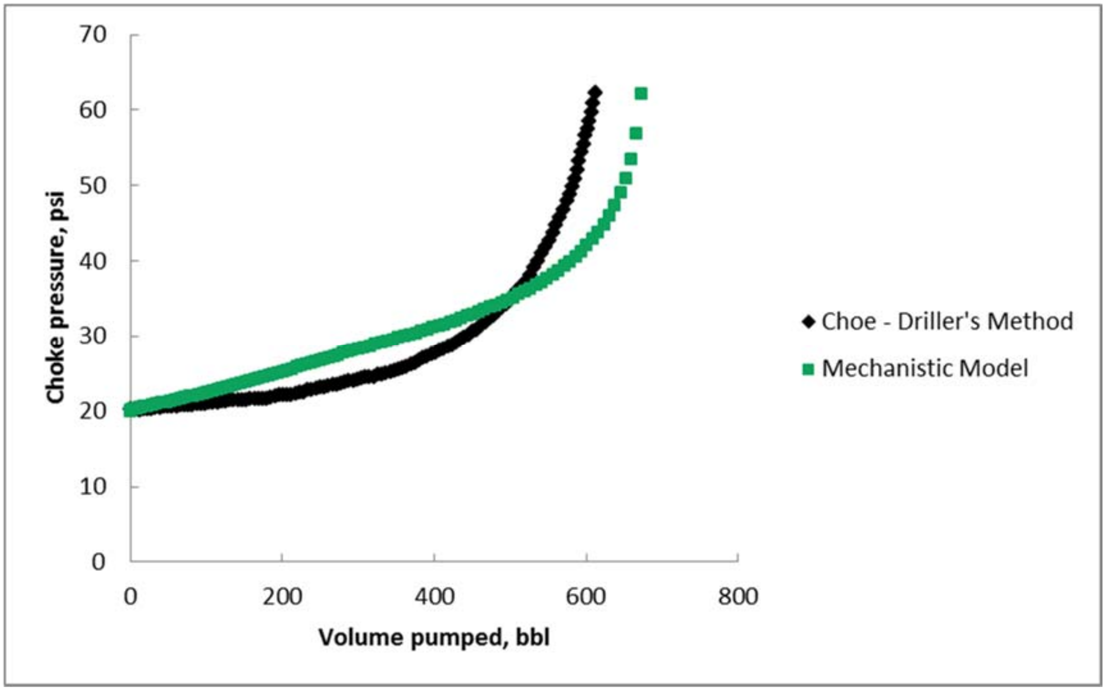

Fig 14 - Comparison between Choe (1995) driller's method and mechanistic model

Fig. 12 shows how the model compares with Santos (1991) model.. A similar comparison is presented between the mechanistic model and the two-phase model by Choe (1995) in Fig. 13. The values for the well-bore geometry, mud flow and mud's properties are the same as shown in Table 4 and Table 5. The rheology data provides for similar rheology of the mud. While both the Santos (1991) model and Choe (1995) model assume a certain temperature gradient per depth, the mechanistic model provides for using the Hasan et al. (1996) model for temperature in the wellbore.

Table 4 – Wellbore geometry and mud flow data for comparison between models

INPUT DATA	
Geometry	
Well Depth (ft):	5400
Depth of casing seat (ft):	2100
Inner diameter of casing (in.):	8.5
Open hole diameter (in.):	8.5
Length of drill-collars (ft):	0
OD drill-collar (in.):	7
ID drill-collar (in.):	2.5
OD drill-pipe (in.):	5
ID drill-pipe (in.):	3.826
Pipe rioughness:	0.00015
Length Increment (ft):	100
Mud flow	
Mud flow rate (gal/min):	150
Mud density (lb/gal):	12
Bit nozzle	
Bit nozzle (32nds in):	12
Bit nozzle 2 (32nds in):	12
Bit nozzle 3 (32nds in):	12

Table 5 – Viscometer reading and temperature characteristics data for comparison between models

INPUT DATA	
Mud viscometer reading	
R3:	3
R100:	20
R300:	39
R600:	65
Temperature data	
Inlet mud temperature, °F	70
Mud thermal conductivity, Btu/(ft-°F-hr)	1
Mud specific heat, Btu/(lbm-°F)	0.4
Formation thermal conductivity, Btu/(ft-°F - hr)	1.3
Formation specific heat, Btu/(lbm-°F)	0.2
Formation density, lbm/ft3	165
Surface earth temperature, °F	70
Geothermal gradient, °F/ft	0.0127
Circulation time, hrs	44

Fig 12 and Fig 13 both show the mechanistic model taking longer to reach the same choke pressure. The main reason being the simulator for mechanistic model takes two-phase flow from depth of entry till gas reaches surface while the model by Santos (1991) and Choe (1995) treat a specific gas volume being circulated to surface.

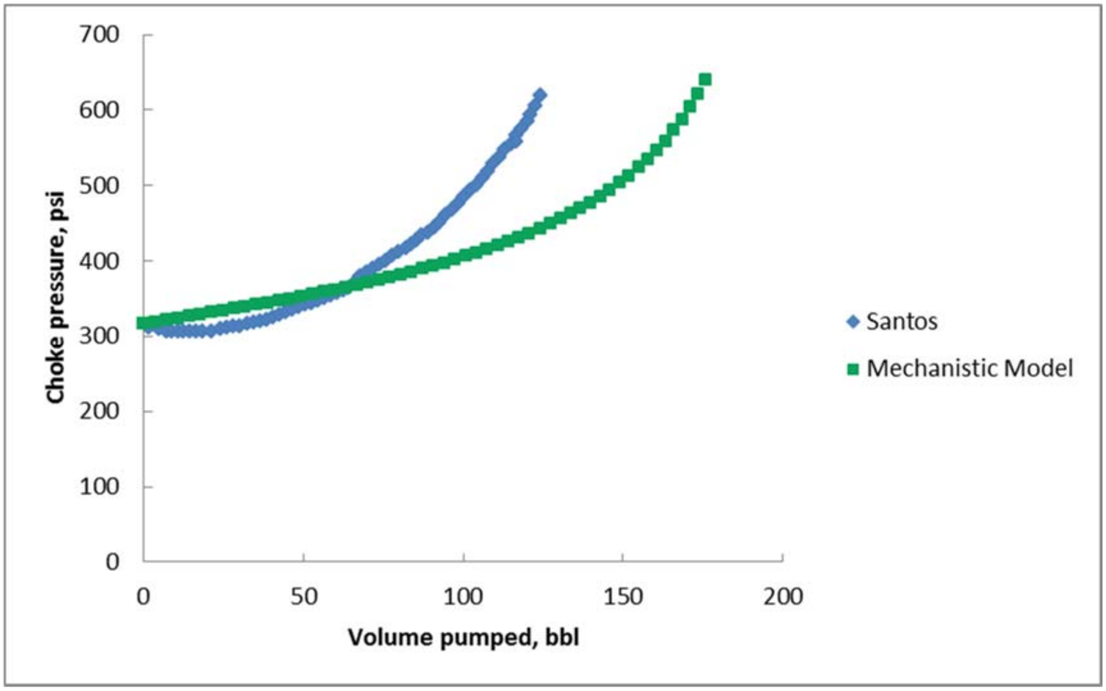

Fig 15 - Comparison between Santos (1991) model and mechanistic model

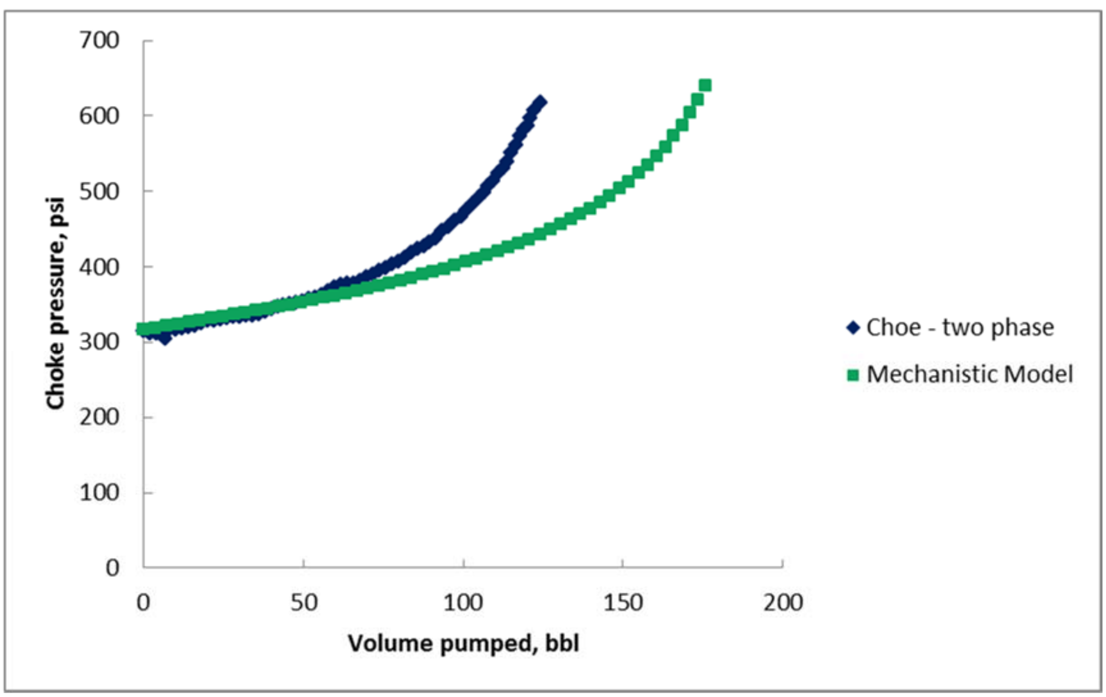

Fig 16 - Comparison between Choe (1995) two phase model and mechanistic model

CONCLUSIONS

This research provides a simulator to model gas kick migration in the annulus as it rises to the surface. The simulator creates an initial model of the mud circulation while drilling in the wellbore prior to the occurrence of the gas kick. The gas kick is simulated by using user specified formation pressure greater than the annulus pressure. The two-phase model incorporates the mechanistic model for two phase flow presented by Hasan and Kabir (2002). This model specifies the flow into different flow regimes to estimate the gas void fraction. The gas void fraction affects all parameter of the two phase flow including pressure drop, mixture density and gas velocity in the annulus, and the estimation of pit gain. The mechanistic approach provides for a more scientific method to estimate gas void fraction as compared to the existing empirical models.

The model is presented in a simulation which shows changes in annulus pressure profile, temperature profile and fluid density as gas migrates to surface. In addition, the simulator shows increasing annulus pressure and pit gain as function of time and mud pumped. The simulator improves on the existing models as providing a detailed analysis in changes of the parameters in the annulus as gas kick migrates up to the surface. The simulator provides for user friendly input sheets and simulated output values and charts in Excel for user to interpret the trend.

RECOMMENDATIONS FOR FUTURE WORK

The simulator could be enhanced by investigating the assumption of constant bottom-hole pressure and accounting for flow parameters at the surface for improved gas kick detection. The simulator could also find applications incorporating sensors to detect gas kicks. The mechanistic model could further be enhanced to circulating a fixed volume of gas out of the wellbore after shut-in using initial pit gain and shut-in pressures.

The temperature model for the two phase flow in the annulus could be further improved by solving for the heat transfer coeffient and accounting for heat transfer through cementing and casing. The temperature model could also incorporate the difference in pressure drops as gas rise up to surface and its influence on the temperature.

The model could also find appplications for deep water drilling, however would have toadjust for flow through the ocean water. The ocean water's influence on the temperature on the flow in the annulus and in the tubing would have to be investigated. This could find more application in cold waters where gas kicks could result in gas hydrates in the wellbore.

The model also assumes water-based mud, and future models could include oil based mud. The oil based mud method would have to account for gas solubility into the mud.

REFERENCES

Avelar, C. S., Riberio, P. R., and Sepehrnoori, K. 2009. Deepwater gas kick simulation. *Journal of Petroleum Science and Engineering* **67**: 13-22.

Aziz, K., Govier, G. W., and Fogarrasi, M. 1972. Pressure Drop in Wells Producing Oil and Gas. *Journal of Canadian Petroleum Technology* (July-Sept) 38-42.

Beggs, D. H. and Brill, J. P. 1973. A Study of Two-Phase Flow in Inclined Pipes. *Journal of Petroleum Technology* **25** (5): 607-617

Chen, N. H. 1979. An Explicit Equation for Friction Factor in Pipe. *Industrial and Engineering Chemistry Research Fundamentals* **18** (3): 296-297

Choe, J. 1995. Dynamic Well Control Simulation Models for Water-based Muds and their Computer Applications. PhD dissertation, Texas A&M University, College Station, Texas (May 1995).

Economides, M. J., Hill, D. A., Economides, C. E. and Zhu, D. 2013. *Petroleum Production Systems*, 61-94. Upper Saddle River, New Jersey, Pearson Education.

Harmathy, T. Z. 1960. Velocity of large drops and bubbles in media of infinite or restricted extent. *AIChE Journal* **6** (2): 281-288

Hasan, A.R. and Kabir, C.S. 1986. A Study of Multiphase How Behavior in Vertical Oil Wells: Part I - Theoretical Treatment. Paper SPE 15138 presented at the 56th California Regional Meeting of the SPE, Oakland, California, 2–4 April.

Hasan, A.R., and Kabir, C. S. 1992. Two-phase Flow in Vertical and Inclined Annuli. *International Journal of Multiphase Flow* **18** (2): 279-293

Hasan, A.R. and Kabir, C.S. 2002. *Fluid Flow and Heat Transfer in Wellbores*, 18–60. Richardson, Texas, SPE.

Hasan, A.R., Kabir, C.S., Ameen, M. M. and Wang, X. 1996. A Mechanistic Model for Circulating Fluid Temperature. *SPE Journal* **1** (2): 133-144

LeBlanc, J. L. and Lewis, R. L. 1968. A Mathematical Model of a Gas Kick. *Journal of Petroleum Technology* **20** (8): 888-898

Nickens, H. V. 1987. A Dynamic Computer Model of a Kicking Well. *SPE Drilling Engineering* **2** (2): 159-173.

Nunes, J. O. L., Bannwart, A.C., and Ribeiro, P.R. 2002. Mathematical Modeling of Gas Kicks in Deep Water Scenario. Paper SPE 16011 presented at the IADC/SPE Asia Pacific Drilling Technology, Jakarta, Indonesia, 8–11 September.

Omosebi, A. O., Osisanya, S. O., Chukwu, G. A. and Egbon, F. 2012. Annular Pressure Prediction During Well Control. Paper SPE 163015 presented at the Nigeria Annual International Conference and Exhibition, Lagos, Nigeria, 6-8 August

Rader, D. W., Bourgoyne Jr, A. T., and Ward, R. H. 1975. Factors Affecting Bubble-Rise Velocity Of Gas Kicks. *Journal of Petroleum Technology* **27** (5): 571-584

Santos, O. L. A. 1991. Well-Control Operations in Horizontal Wells. *SPE Drilling Engineering* **6** (2): 111-117.

Starrett, M. P., Hill, D. A., and Sepehrnoori, K. 1990. A Shallow-Gas-Kick Simulator Including Diverter Performance. *SPE Drilling Engineering* **5** (1): 79-85.

United States Department of Labor, Occupational Safety and Health Administration. 2003. Oil and Gas well drilling and Servicing etool.

https://www.osha.gov/SLTC/etools/oilandgas/drilling/wellcontrol.html (accessed 03 March 2014)

Wallis, G.B. 1969. *One Dimensional Two-Phase Flow*, 93-100, New York City, New York, McGraw-Hill Book Co. Inc.

Zuber, N. and Findlay, J. 1965. Average Volumetric Concentration in Two-Phase Flow Systems. *ASME J. Heat Transfer* **87**: 453

APPENDIX A

EQUATIONS

<u>Power Law method</u>

The Power Law method provides for calculating the effective viscosity of the mud circulating in the tubing and the annulus. The calculation in tubing use viscometer reading at 600 RPM and 300 RPM.

$$n_t = 3.32\, log\left(\frac{R_{600}}{R_{300}}\right) \tag{1}$$

$$K_t = \frac{5.11 R_{600}}{1022^{n_t}} \tag{2}$$

$$V_t = \frac{0.408 q_{mud}}{{d_H}^2} \tag{3}$$

$$\mu_{et} = 100\, K \left(\frac{96V}{d_H}\right)^{n_t - 1} \left(\frac{3n_t + 1}{4n_t}\right)^{n_t} \tag{4}$$

The calculation of effective viscosity using the Power Law method in the annulus use viscometer reading at 100 RPM and 3 RPM. The equations used to calculate the effective viscosity are given as:

$$n_a = 0.657\, log\left(\frac{R_{100}}{R_3}\right) \tag{5}$$

$$K_a = \frac{5.11 R_{100}}{170.2^{n_a}} \tag{6}$$

$$V_a = \frac{0.408 q_{mud}}{{d_H}^2} \tag{7}$$

$$\mu_{ea} = 100\ K \left(\frac{144V}{d_H}\right)^{n_a - 1} \left(\frac{2n_a + 1}{3n_a}\right)^{n_a} \tag{8}$$

<u>Reynolds's number and friction factor</u>

The Reynolds's number is used to calculate the frictional pressure drop in the wellbore is calculated as:

$$N_{Re} = \frac{d_H V \rho_{Mud}}{\mu_e} \tag{9}$$

The fanning friction factor is calculated using Chen's equation (Chen, 1979) given in Eq.10. The Moody friction factor which is used in the pressure drop calculation is four times the fanning friction factor.

$$\frac{1}{\sqrt{f_f}} = -4 \log\left\{\frac{\mathcal{E}}{3.7065} - \frac{5.0452}{N_{Re}} \log\left[\frac{(\mathcal{E})^{1.1098}}{2.8257} + \left(\frac{7.149}{N_{Re}}\right)^{0.8981}\right]\right\} \tag{10}$$

<u>Single phase pressure model</u>

The single phase pressure model calculates the pressure in the tubing by adding the hydrostatic pressure (Eq. 11) and subtracting the frictional pressure drop (Eq. 12) at each increment. While in the annulus the total pressure drop accounts both the hydrostatic pressure drop and the frictional pressure drop since the fluid is flowing upwards.

$$\left(\frac{dp}{dz}\right)_H = \rho_{Mud} * 0.052 \tag{11}$$

$$\left(\frac{dp}{dz}\right)_F = \frac{fv^2\rho_{mud}}{25.81d_H} \tag{12}$$

The pressure loss at the bit is calculated using Eq. 3.

$$\mathrm{p}_{bit} = \frac{156\rho_{mud}q_{mud}}{(bn1^2 + bn2^2 + bn3^2)^2} \tag{13}$$

Gas deviation factor

The calculation for deviation factor Z use the equations presented by Economides et. al (2013). The calculation depends on pseudo-reduced property of pressure and temperature which are given in Eq. 16 and Eq. 17. The pseudo-critical pressure and temperature are given in Eq.14 and Eq. 15.

$$p_{pc} = 677 + 15\gamma_g - 35.7{\gamma_g}^2 \tag{14}$$

$$T_{pc} = 168 + 325\gamma_g - 12.5{\gamma_g}^2 \tag{15}$$

$$p_{pr} = \frac{p}{p_{pc}} \tag{16}$$

$$T_{pr} = \frac{T}{T_{pc}} \tag{17}$$

The calculation of gas deviation factor Z use an iterative loop since Z depends of ρ_{pr} (Eq. 18) while ρ_{pr} depends on Z (Eq.19).

$$Z = 1 + (A_1 + \frac{A_2}{T_{pr}} + \frac{A_3}{T^3{}_{pr}} + \frac{A_4}{T^4{}_{pr}} + \frac{A_5}{T^5{}_{pr}})\rho_{pr}) + (A_6 + \frac{A_7}{T_{pr}} + \frac{A_8}{T_{pr}^2})\rho^2{}_{pr} - A_9(\frac{A_7}{T_{pr}} + \frac{A_8}{T_{pr}^2})(\rho^5{}_{pr}) + A_{10}(1 + A_{11}\rho^2{}_{pr})(\frac{\rho^2{}_{pr}}{T_{pr}^3})\,EXP(-A_{11}\rho^2{}_{pr}) \tag{18}$$

Where $\rho_{pr} = 0.27\left[\frac{p_{pr}}{(ZT_{pr})}\right]$ (19)

The constants are as follows: A_1 = 0.3265, A_2 = -1.0700, A_3 = -0.5339, A_4 = 0.01569, A_5 = -0.05165, A_6 = 0.5475, A_7 = -0.7361, A_8 = 0.1844, A_9 = 0.1056, A_{10} = 0.6134, A_{11} = 0.7210

Gas Viscosity Calculation

The gas viscosity is calculated using Eq. 20 as presented by Economides at. al (2013).

$$\mu_g = A_a(10^{-4})EXP(B_b {\rho_g}^{C_c}) \tag{20}$$

Where

$$A_a = \frac{(9.379 + 0.01607M_a)T^{1.5}}{209.2 + 19.26M_a + T} \tag{21}$$

$$B_b = 3.448 + \frac{986.4}{T} + 0.01009M_a \tag{22}$$

$$C_c = 2.447 - 0.2224B_b \tag{23}$$

Here the apparent molecular weight is given as:

$$M_a = 28.97\gamma_g \tag{24}$$

Gas influx

The gas influx into the annulus is calculated using Darcy flow through the pressure difference between the formation and the wellbore.

$$q_g = \frac{kh\,({p_r}^2 - {p_a}^2)}{(1424\mu_g ZTln\left(\frac{r_e}{r_w}\right))} \tag{25}$$

Two phase pressure drop calculations

The two phase calculations of the pressure drop use the mechanistic method presented by Hasan and Kabir (2002). The model presents five flow regimes for the two-phase flow as bubbly, slug, churn, dispersed bubbly and annular which incorporates the pressure drop as the flow behavior changes. The pressure drop calculation relies on as accurate measurement of the in-situ gas void fraction in order to calculate the pressure gradient through the annulus (Hasan and Kabir, 1992). The expression for gas void pressure is given by:

$$f_g = \frac{v_{sg}}{C_o v_m + \, v_\infty} \tag{26}$$

The value of flow parameter C_0 is given as 1.2 for bubbly, slug and dispersed bubbly flow, 1.15 for churn flow and 1.0 for annular flow. The single bubble terminal rise velocity, v_∞ is calculated using the expression presented by Harmathy (1960).

$$v_\infty = 1.53 \left[\frac{g(\rho_L - \rho_g)\sigma}{{\rho_L}^2}\right]^{1/4} \tag{27}$$

The transition from bubbly to slug flow is critical for calculating the two phase pressure drop during a gas kick into the annulus. Bubbly flow occurs when f_g is less than or equal 0.25 and after that the transition to slug flow occurs when v_t is greater than v_{sg}, the superficial velocity of gas. Here, v_t is given by

$$v_t = 0.429\, v_{sg} + 0.357\, v_\infty \tag{28}$$

The slug flow is characterized as a large Taylor bubble followed by small bubble. The Taylor bubble velocity is given by

$$v_{\infty T} = 0.35 \sqrt{\frac{gd(\rho_L - \rho_g)}{\rho_L}} \tag{29}$$

For slug flow, the Taylor bubble flow must also be greater than the bubble velocity, as the slug moves up taking up the smaller bubble. The gas void fraction takes an expression using average bubble velocity during slug flow.

$$f_g = \frac{v_{sg}}{C_o v_m + \bar{v}_\infty} \tag{30}$$

The average bubble velocity given by the expression

$$\bar{v}_\infty = v_\infty\left(1 - e^{-v_t/v_{sg}}\right) + v_{\infty T} e^{-v_t/v_{sg}} \tag{31}$$

The other transition includes transition to dispersed bubbly when the mixture velocity is great. The mixture velocity must exceed the expression shown in Eq. 32 for dispersed bubbly flow to exist.

$$v_m > \left[\frac{1}{2}\left(0.725 + 4.15\sqrt{\frac{v_{sg}}{v_m}}\right)\sqrt{\frac{g(\rho_L - \rho_g)}{0.4\sigma}}\ \left(\frac{\sigma}{\rho_L}\right)^{0.6}\ \left(\frac{2d}{f}\right)^{0.4}\right]^{0.833} \tag{32}$$

For churn flow to exist, the gas velocity must be greater a certain magnitude than the liquid velocity as shown in Eq. 33.

$$v_{sg} > 1.08\, v_{sl} \tag{33}$$

Annular flow exists with large amount of gas flow with f_g exceeding 0.85. The v_{sg} must exceed v_{ann}, given by Eq. 34.

$$v_{ann} = 3.1\ \left[\frac{g\sigma(\rho_L - \rho_g)}{{\rho_g}^2}\right]^{1/4} \tag{34}$$

The pressure drop calculation for annular and frictional pressure drop relies heavily on the mixture density. The mixture density would also be a criteria for gas kick detection, depends on the gas void fraction. The expression for mixture density is given as

$$\rho_m = f_g \rho_g + (1 - f_g)\rho_l \tag{35}$$

To calculate mixture viscosity (Eq. 16), the gas mass fraction is used given by Eq. 15.

$$x = \frac{v_{sg}\rho_g}{(v_{sg}\rho_g + v_{sl}\rho_l)} \tag{36}$$

$$\mu_m = x\mu_g + (1 - x)\mu_l \tag{37}$$

The two phase hydrostatic and frictional pressure gradient is then given by Eq. 38 and Eq. 39.

$$\left(\frac{dp}{dz}\right)_H = \rho_m \left(\frac{1}{144}\right) \tag{38}$$

$$\left(\frac{dp}{dz}\right)_F = \left(\frac{1}{144}\right)\frac{f {v_m}^2 \rho_m}{2 g_c d} \tag{39}$$

<u>Temperature model calculation</u>

The temperature calculation from Hasan and Kabir (1996) in the drillpipe and annulus is given by the following expressions.

$$T_t = \alpha e^{\lambda_1 z} + \beta e^{\lambda_2 z} + g_G z - B g_G + T_{es} \quad (40)$$

$$T_a = (1 + B\lambda_1)\alpha e^{\lambda_1 z} + (1 + B\lambda_2)\beta e^{\lambda_2 z} + g_G z + T_{es} \quad (41)$$

Here B is given by

$$B = \frac{w c_{fl}}{2\pi r_t U_t} \quad (42)$$

Here, the constants α, β, λ_1, λ_2 are given by the following expression

$$\alpha = -\frac{(T_{ti} - T_{es} + B g_G)\lambda_2 e^{\lambda_2 L} + g_G}{\lambda_1 e^{\lambda_1 L} - \lambda_2 e^{\lambda_2 L}} \quad (43)$$

$$\beta = \frac{(T_{ti} - T_{es} + B g_G)\lambda_1 e^{\lambda_1 L} + g_G}{\lambda_1 e^{\lambda_1 L} - \lambda_2 e^{\lambda_2 L}} \quad (44)$$

$$\lambda_1 = \frac{L_R}{2} + \frac{L_R}{2}\sqrt{1 + 4(r_c U_a T_D + k_e)\frac{r_t U_t}{r_c U_a k_e}} \quad (45)$$

$$\lambda_2 = \frac{L_R}{2} - \frac{L_R}{2}\sqrt{1 + 4(r_c U_a T_D + k_e)\frac{r_t U_t}{r_c U_a k_e}} \tag{46}$$

The relaxation parameter L_R is defined by

$$L_R = \frac{2\pi}{c_{fl} w}\left[\frac{r_c U_a T_D}{k_e + r_c U_a k_e}\right] \tag{47}$$

Temperature model modification for two-phase

The temperature calculations for the two-phase use the average specific heat of the mixture is computed from the specific heat and flow of both the phases. The mass rate for the two-phase incorporates the mixture velocity and density flowing through the annulus area.

$$c_{avg} = \frac{\left(c_{fl} q_{mud} \rho_{mud}\right) + \left(c_g q_g \rho_g\right)}{\left(q_{mud} \rho_{mud}\right) + \left(q_g \rho_g\right)} \tag{48}$$

$$w = v_m \rho_m (Area) \tag{49}$$

The temperature addition from the influx is shown

$$E_{entry} = \left(c_{fl} q_{mud} \rho_{mud} T_{mud}\right) + \left(c_g q_g \rho_g T_g\right) \tag{50}$$

$$T_{mixture} = \frac{E_{entry}}{\left((q_{mud}\rho_{mud}) + (q_g\rho_g)\right)c_{avg}} \tag{51}$$

$$T_{addition} = T_{mixture} - T_{mud} \tag{52}$$

APPENDIX B

NOMENCLATURE

A_1 = gas deviation calculation constant, dimensionless

A_2 = gas deviation calculation constant, dimensionless

A_3 = gas deviation calculation constant, dimensionless

A_4 = gas deviation calculation constant, dimensionless

A_5 = gas deviation calculation constant, dimensionless

A_6 = gas deviation calculation constant, dimensionless

A_7 = gas deviation calculation constant, dimensionless

A_8 = gas deviation calculation constant, dimensionless

A_9 = gas deviation calculation constant, dimensionless

A_{10} = gas deviation calculation constant, dimensionless

A_{11} = gas deviation calculation constant, dimensionless

A_a = constant defined by Eq. 21, dimensionless

B = constant defined by Eq. 42, dimensionless

B_b = constant defined by Eq. 22, dimensionless

bn1 = bit nozzle 1 diameter, 32^{nds} in.

bn2 = bit nozzle 2 diameter, 32^{nds} in.

bn3 = bit nozzle 3 diameter, 32^{nds} in.

c_{avg} = specific heat of mixture, Btu/(lbm-°F)

c_{fl} = specific heat of fluid, Btu/(lbm-°F)

c_g = specific heat of gas, Btu/(lbm-°F)

C_0 = flow parameter in bubbly flow, dimensionless

C_c = constant defined by Eq. 23, dimensionless

d_H = hydraulic diameter, in.

$(dp/dz)_H$ = hydro-static pressure gradient, psi/ft

$(dp/dz)_F$ = frictional pressure drop pressure gradient, psi/ft

E_{energy} = heat energy at the point of gas entry, J

f_f = fanning friction factor, dimensionless

f = Moody friction factor, dimensionless

f_g = volumetric in-situ gas fraction

g = acceleration owing to gravity, ft/sec^2

g_c = conversion factor, 32.17 lbm-ft/lbf-sec^2

h = reservoir height, ft

k = formation permeability, mD

K_a = fluid consistency index in annulus, dimensionless

K_t = fluid consistency index in tubing, dimensionless

k_e = formation conductivity, Btu / (hr-ft-°F)

L = total vertical well depth, ft

L_R = relaxation distance parameter, 1/ft

M_a = apparent molecular weight, g/mol

n_a = power law constant for flow in annulus, dimensionless

n_t = power law constant for flow in tubing, dimensionless

p = pressure of gas, psi

p_a = annulus pressure, psi

p_{bit} = pressure loss at the bit, psi

p_{pc} = pseudo-critical pressure of gas, psi

p_{pr} = pseudo-reduced property of pressure, dimensionless

p_r = reservoir pressure, psi

q_{mud} = mud flow rate, gal/min

q_g = gas flow into the wellbore, Mscf/D

r_c, r_w = wellbore radius, ft

r_e = reservoir radius, ft

r_t = drillpipe radius, ft

R3 = rotational viscometer readings at 3 RPM, dimensionless

R100 = rotational viscometer readings at 100 RPM, dimensionless

R300 = rotational viscometer readings at 300 RPM, dimensionless

R600 = rotational viscometer readings at 600 RPM, dimensionless

T = temperature of gas, °F

T_a = annulus fluid temperature, °F

$T_{addition}$ = addition of temperature from gas influx, °F

T_D = dimensionless temperature

T_{es} = formation static temperature at surface, °F

$T_{mixture}$ = temperature of the mixture from the heat energy at gas entry, °F

T_{pc} = pseudo-critical temperature of gas, °F

T_{pr} = pseudo-reduced property of temperature, dimensionless

T_t = drillpipe fluid temperature, °F

T_{ti} = fluid temperature entering the wellbore, °F

U_a = overall heat-transfer coefficient for annulus, Btu/(hr-ft^2-°F)

U_t = overall heat-transfer coefficient for drillpipe, Btu/(hr-ft^2-°F)

v_∞ = terminal rise velocity of single bubble, ft/sec

$v_{\infty T}$ = rise velocity of Taylor bubble, ft/sec

v_{sg} = superficial gas in-situ velocity, ft/sec

v_{sl} = superficial liquid in-situ velocity, ft/sec

v_m = mixture velocity, ft/sec

v_t = superficial gas velocity needed for transition from bubbly to slug flow, ft/sec

V_a = average bulk velocity in annulus, ft/sec

V_t = average bulk velocity in tubing, ft/sec

w = mass flow rate of fluid, lbm/hr

x = gas mass fraction of fluid, lbm/hr

Z = gas-law deviation factor, dimensionless

α = constant defined by Eq. 43, dimensionless

β = = constant defined by Eq. 44, dimensionless

λ_1 = constant defined by Eq. 45, dimensionless

ε = relative pipe roughness, dimensionless

λ_2 = constant defined by Eq. 46

μ_g = gas viscosity, cp

μ_l = liquid viscosity, cp

μ_m = mixture viscosity, cp

ρ_g = gas density, lbm/ft^3 p

ρ_l = liquid density, lbm/ft^3

ρ_m = mixture density, lbm/ft^3

ρ_{mud} = mud density, lbm/gal

σ = surface tension, lbm/sec^2

γ_g = gas gravity, dimensionless

APPENDIX C

SIMULATOR CODE

The VBA code for mud-circulation simulation:

```
Private Sub CommandButton1_Click()
''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

'clear columns on the sheet for data
Sheets("MC - Profiles").Cells.Range("A7:J1000").ClearContents
Sheets("MC - Profiles").Cells.Range("B7:K1000").ClearContents
Sheets("MC - Profiles").Cells.Range("C7:K1000").ClearContents
Sheets("MC - Profiles").Cells.Range("D7:J1000").ClearContents
Sheets("MC - Profiles").Cells.Range("E7:K1000").ClearContents

Sheets("Temp Data-MC").Cells.Range("A3:O1000").ClearContents
Sheets("MC-Data").Cells.Range("A3:O1000").ClearContents

'collect values from the excel sheet
'collect geometry data from the 'Mud Circulation' sheet
Dim well_depth, CSG_depth, ID_casing, OH_diameter, DC_length, OD_DC, ID_DC, OD_DP,
ID_DP, pipe_roughness As Double

well_depth = Range("E6").Value
CSG_depth = Range("E7").Value
ID_casing = Range("E8").Value
OH_diameter = Range("E9").Value
DC_length = Range("E10").Value
OD_DC = Range("E11").Value
ID_DC = Range("E12").Value
OD_DP = Range("E13").Value
ID_DP = Range("E14").Value
pipe_roughness = Range("E15").Value

'check to see any invalid inputs
If OD_DC < ID_DC Or OH_diameter < OD_DC Or OH_diameter < ID_DC Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If

If OD_DP < ID_DP Or OH_diameter < OD_DP Or OH_diameter < ID_DP Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If

If well_depth < CSG_depth Or well_depth < DC_length Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
```

```
End If

If Not IsNumeric(OD_DC) Or Not IsNumeric(ID_DC) Or Not IsNumeric(OD_DP) Or Not
IsNumeric(ID_DP) Or Not IsNumeric(OH_diameter) Or Not IsNumeric(ID_casing) Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If

If Not IsNumeric(well_depth) Or Not IsNumeric(CSG_depth) Or Not IsNumeric(DC_length) Or
Not IsNumeric(pipe_roughness) Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If


'collect length increment from the 'Mud Circulation' sheet
Dim length_inc As Double
length_inc = Range("E17").Value

'check to see any invalid inputs
If length_inc > well_depth Or length_inc > CSG_depth Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If

If Not IsNumeric(length_inc) Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If


'collect mud density and mud flow from the 'Mud Circulation' sheet
Dim mud_flow_rate, mud_density As Double
mud_flow_rate = Range("E20").Value
mud_density = Range("E21").Value

'collect bit nozzle info from the 'Mud Circulation' sheet
Dim bit_nozzle1, bit_nozzle2, bit_nozzle3 As Double
bit_nozzle1 = Range("E24").Value
bit_nozzle2 = Range("E25").Value
bit_nozzle3 = Range("E26").Value

'collect mud viscometer readings from the 'Mud Circulation' sheet
Dim R3, R100, R300, R600 As Double
R3 = Range("H6").Value
R100 = Range("H7").Value
R300 = Range("H8").Value
R600 = Range("H9").Value

'collect values from the excel sheet for temperature calculations
Dim inlet_mud_temp, mud_k, mud_spec_heat As Double
Dim formation_k, formation_spec_heat, formation_density As Double
Dim surf_earth_temp, geo_grad, time As Double
```

```
inlet_mud_temp = Range("H12").Value
mud_k = Range("H13").Value
mud_spec_heat = Range("H14").Value
formation_k = Range("H15").Value
formation_spec_heat = Range("H16").Value
formation_density = Range("H17").Value
surf_earth_temp = Range("H18").Value
geo_grad = Range("H19").Value
time = Range("H20").Value

'initialize array length
Dim end_length As Double
Dim Num_WD_value As Integer
end_length = well_depth Mod length_inc
Num_WD_value = ((well_depth - end_length) / length_inc)
Array_length = Num_WD_value + 4

'calculate hole_diameter, tubing_ID, tubing_OD through the wellbore
Dim hole_diameter(), tubing_ID(), tubing_OD(), depth_array() As Double
ReDim depth_array(1 To Array_length)
ReDim hole_diameter(1 To Array_length)
ReDim tubing_ID(1 To Array_length)
ReDim tubing_OD(1 To Array_length)

'update arrays with the geometry and display results in MC-Data sheet
Call Geometry(depth_array, hole_diameter, tubing_ID, tubing_OD, well_depth, CSG_depth,
ID_casing, OH_diameter, DC_length, OD_DC, ID_DC, OD_DP, ID_DP, length_inc)

'initalize variables for flow in tubing
Dim visc_tubing, dpF_dL_tubing, dpF_dL_tubing_inc, dpF_total_tubing As Double
Dim visc_annulus, dpF_dl_annulus, dpF_dL_annulus_inc, dpF_total_annulus As Double
Dim act_length_inc As Double

'calculate total frictional pressure drop in tubing
dpF_total_tubing = 0
For i = 2 To Array_length

act_length_inc = depth_array(i) - depth_array(i - 1)
Call DP_Frictional_tubing(R300, R600, mud_flow_rate, mud_density, pipe_roughness,
tubing_ID(i), visc_tubing, dpF_dL_tubing)
dpF_dL_tubing_inc = dpF_dL_tubing * act_length_inc
dpF_total_tubing = dpF_total_tubing + dpF_dL_tubing_inc

Next i

'calculate total frictional pressure drop in annulus
dpF_total_annulus = 0
For i = Array_length - 1 To 1 Step -1

act_length_inc = depth_array(i + 1) - depth_array(i)
Call DP_Frictional_annulus(R100, R3, mud_flow_rate, mud_density, pipe_roughness,
hole_diameter(i), tubing_OD(i), visc_annulus, dpF_dl_annulus)
```

```
dpF_dL_annulus_inc = dpF_dl_annulus * act_length_inc
dpF_total_annulus = dpF_total_annulus + dpF_dL_annulus_inc

Next i

'calculate pressure drop at the bit
Dim dp_bit_nozzle As Double
dp_bit_nozzle = (156 * mud_density * mud_flow_rate ^ 2) / ((bit_nozzle1 ^ 2 + bit_nozzle2 ^ 2 +
bit_nozzle3 ^ 2) ^ 2)
Sheets("MC - Profiles").Cells(4, 13) = dp_bit_nozzle

'display SCP, tubing viscosity, annulus viscosity
Dim SCP, initial_standpipe_pressure As Double

initial_standpipe_pressure = dpF_total_tubing + dp_bit_nozzle + dpF_total_annulus
Sheets("MC - Profiles").Cells(4, 3) = initial_standpipe_pressure
Sheets("MC - Profiles").Cells(4, 6) = visc_tubing
Sheets("MC - Profiles").Cells(4, 10) = visc_annulus

'create arrays for pressure profile in tubing and annulus
Dim press_tub_array(), press_ann_array() As Double
ReDim press_tub_array(1 To Array_length)
ReDim press_ann_array(1 To Array_length)


'intialize first array elements for pressure profile in tubing
press_tub_array(1) = initial_standpipe_pressure
Sheets("MC - Profiles").Cells(7, 1) = 0
Sheets("MC - Profiles").Cells(7, 2) = press_tub_array(1)
Sheets("MC-Data").Cells(1 + 2, 9) = press_tub_array(1)
Call DP_Frictional_tubing(R300, R600, mud_flow_rate, mud_density, pipe_roughness,
tubing_ID(1), visc_tubing, dpF_dL_tubing)
Sheets("MC-Data").Cells(1 + 2, 5) = visc_tubing

'calulate and display pressure profile in the tubing
Dim dp_hydrostatic, dp_tubing, display As Double
display_ctr = 0

For i = 2 To Array_length

act_length_inc = depth_array(i) - depth_array(i - 1)
Call DP_Frictional_tubing(R300, R600, mud_flow_rate, mud_density, pipe_roughness,
tubing_ID(i), visc_tubing, dpF_dL_tubing)

dpF_tubing = dpF_dL_tubing * act_length_inc
dp_hydrostatic = 0.052 * act_length_inc * mud_density

dp_tubing = dp_hydrostatic - dpF_tubing
press_tub_array(i) = press_tub_array(i - 1) + dp_tubing

If depth_array(i) Mod length_inc = 0 Or depth_array(i) = well_depth Then
If depth_array(i) <> depth_array(i - 1) Then
Sheets("MC - Profiles").Cells(display_ctr + 8, 1) = depth_array(i)
```

```
Sheets("MC - Profiles").Cells(display_ctr + 8, 2) = press_tub_array(i)
display_ctr = display_ctr + 1
End If
End If

Sheets("MC-Data").Cells(i + 2, 5) = visc_tubing
Sheets("MC-Data").Cells(i + 2, 6) = dpF_tubing
Sheets("MC-Data").Cells(i + 2, 7) = dp_hydrostatic
Sheets("MC-Data").Cells(i + 2, 8) = dp_tubing
Sheets("MC-Data").Cells(i + 2, 9) = press_tub_array(i)

Next i

Sheets("MC-Data").Cells(1, 27) = display_ctr

'intialize first array elements for pressure profile in annulus
press_ann_array(Array_length) = press_tub_array(Array_length) - dp_bit_nozzle
display_ctr = display_ctr - 1
Sheets("MC - Profiles").Cells(display_ctr + 8, 3) = press_ann_array(Array_length)
Sheets("MC-Data").Cells(Array_length + 2, 14) = press_ann_array(Array_length)
Call DP_Frictional_annulus(R100, R3, mud_flow_rate, mud_density, pipe_roughness,
hole_diameter(Array_length), tubing_OD(Array_length), visc_annulus, dpF_dl_annulus)
Sheets("MC-Data").Cells(Array_length + 2, 10) = visc_annulus

'calulate and display pressure profile in the annulus
Dim dp_annulus As Double

display_ctr = display_ctr - 1
For i = Array_length - 1 To 1 Step -1

act_length_inc = depth_array(i + 1) - depth_array(i)
Call DP_Frictional_annulus(R100, R3, mud_flow_rate, mud_density, pipe_roughness,
hole_diameter(i), tubing_OD(i), visc_annulus, dpF_dl_annulus)

dpF_annulus = dpF_dl_annulus * act_length_inc
dp_hydrostatic = 0.052 * act_length_inc * mud_density

dp_annulus = dpF_annulus + dp_hydrostatic
press_ann_array(i) = press_ann_array(i + 1) - dp_annulus

If depth_array(i) Mod length_inc = 0 Or depth_array(i) = well_depth Then
If depth_array(i) <> depth_array(i + 1) Then
Sheets("MC - Profiles").Cells(display_ctr + 8, 3) = press_ann_array(i)
display_ctr = display_ctr - 1
End If
End If

Sheets("MC-Data").Cells(i + 2, 10) = visc_annulus
Sheets("MC-Data").Cells(i + 2, 11) = dpF_annulus
Sheets("MC-Data").Cells(i + 2, 12) = dp_hydrostatic
Sheets("MC-Data").Cells(i + 2, 13) = dp_annulus
Sheets("MC-Data").Cells(i + 2, 14) = press_ann_array(i)
```

```
Next i

'create arrays for temperautre profile in tubing and annulus
Dim temp_tub_array(), temp_ann_array() As Double
ReDim temp_tub_array(1 To Array_length)
ReDim temp_ann_array(1 To Array_length)
display_ctr = 0

'calculate temperature profile in tubing and annulus
Dim Z, Tei, temp_tub, temp_ann As Double

For i = 1 To Array_length

'the temperature model assumus constant tubing OD and hold diameter therefore using drill pipe OD and open hole diameter
Z = depth_array(i)
'Call Temp_calc(well_depth, tubing_ID(i), tubing_OD(i), hole_diameter(i), mud_flow_rate, inlet_mud_temp, visc_tubing, visc_annulus, mud_k, mud_spec_heat, mud_density, formation_k, formation_spec_heat, formation_density, surf_earth_temp, geo_grad, time, Z, temp_ann, temp_tub, i, Tei)
Call Temp_calc(well_depth, ID_DP, OD_DP, OH_diameter, mud_flow_rate, inlet_mud_temp, visc_tubing, visc_annulus, mud_k, mud_spec_heat, mud_density, formation_k, formation_spec_heat, formation_density, surf_earth_temp, geo_grad, time, Z, temp_ann, temp_tub, i, Tei)

temp_tub_array(i) = temp_tub
temp_ann_array(i) = temp_ann

'display the temperature profile on worksheet
If depth_array(i) Mod length_inc = 0 Or depth_array(i) = well_depth Then
If i > 1 Then
If depth_array(i) <> depth_array(i - 1) Then
Sheets("MC - Profiles").Cells(display_ctr + 7, 4) = temp_tub_array(i)
Sheets("MC - Profiles").Cells(display_ctr + 7, 5) = temp_ann_array(i)
Sheets("MC - Profiles").Cells(display_ctr + 7, 6) = Tei
display_ctr = display_ctr + 1
End If
Else
Sheets("MC - Profiles").Cells(display_ctr + 7, 4) = temp_tub_array(i)
Sheets("MC - Profiles").Cells(display_ctr + 7, 5) = temp_ann_array(i)
Sheets("MC - Profiles").Cells(display_ctr + 7, 6) = Tei
display_ctr = display_ctr + 1
End If
End If

Next i

End Sub
Sub DP_Frictional_tubing(R300, R600, mud_flow_rate, mud_density, pipe_roughness, tubing_diameter, visc_tubing, dpF_dL_tubing)

Dim n, k, V, N_reg, f As Double
Dim r_r, Re_m, f_m As Double
```

```
'calculate viscosity using the power law method
n = 3.32 * Log10(R600 / R300)
k = (5.11 * R600) / (1022 ^ n)
V = (0.408 * mud_flow_rate) / (tubing_diameter ^ 2)
visc_tubing = (100 * k) * ((96 * V) / tubing_diameter) ^ (n - 1) * ((3 * n + 1) / (4 * n)) ^ n

'calculate frictional pressure drop gradient using chen's  friction factor calculation
N_reg = (928 * tubing_diameter * mud_density * V) / visc_tubing

r_r = pipe_roughness / (tubing_diameter / 12)
Re_m = ((tubing_diameter / 12) * V * mud_density * 7.4805) / (visc_tubing * 0.000672)
f_m = 4 / ((4 * Log10((r_r / 3.7065) - ((5.0452 / Re_m) * Log10(((r_r ^ 1.1098) / 2.8257) + ((7.149
/ Re_m) ^ 0.8981))))) ^ 2)

f = f_m / 4
dpF_dL_tubing = (f * mud_density * V ^ 2) / (25.81 * tubing_diameter)

End Sub

Sub DP_Frictional_annulus(R100, R3, mud_flow_rate, mud_density, pipe_roughness,
hole_diameter, tubing_OD, visc_annulus, dpF_dl_annulus)

Dim n2, k2, V2, N_reg2, f2 As Double
Dim r_r, Re_m, f_m As Double

'calculate viscosity using the power law method
n2 = 0.657 * Log10(R100 / R3)
k2 = (5.11 * R100) / (170.2 ^ n2)
V2 = (0.408 * mud_flow_rate) / (hole_diameter ^ 2 - tubing_OD ^ 2)
visc_annulus = (100 * k2) * ((144 * V2) / (hole_diameter - tubing_OD)) ^ (n2 - 1) * ((2 * n2 + 1) /
(3 * n2)) ^ n2

'calculate frictional pressure drop gradient using chen's  friction factor calculation
N_reg2 = 928 * (hole_diameter - tubing_OD) * V2 * mud_density / visc_annulus
r_r = pipe_roughness / ((hole_diameter - tubing_OD) / 12)
Re_m = (((hole_diameter - tubing_OD) / 12) * V2 * mud_density * 7.4805) / (visc_annulus *
0.000672)
f_m = 4 / ((4 * Log10((r_r / 3.7065) - ((5.0452 / Re_m) * Log10(((r_r ^ 1.1098) / 2.8257) + ((7.149
/ Re_m) ^ 0.8981))))) ^ 2)

f2 = f_m / 4
dpF_dl_annulus = (f2 * V2 ^ 2 * mud_density) / (25.81 * (hole_diameter - tubing_OD))

End Sub



Sub Geometry(depth_array, hole_diameter, tubing_ID, tubing_OD, well_depth, CSG_depth,
ID_casing, OH_diameter, DC_length, OD_DC, ID_DC, OD_DP, ID_DP, length_inc)

'initialize arrays at surface
depth_array(1) = 0
```

```
hole_diameter(1) = ID_casing
tubing_OD(1) = OD_DP
tubing_ID(1) = ID_DP

'incorporate different depths of casing and drill collar in the length increment by adding variable
to the array
Dim DP_depth As Double
DP_depth = well_depth - DC_length

'calculate additional depth after dividing by length increments
Dim end_length, end_CSG_length, end_DP_length As Double
end_length = well_depth Mod length_inc
end_CSG_length = CSG_depth Mod length_inc
end_DP_length = DP_depth Mod length_inc

'calculate number of increments at each depth
Dim Num_WD_value, Num_CSD_value, Num_DPD_value, Array_length As Integer
Num_WD_value = ((well_depth - end_length) / length_inc)
Num_CSD_value = ((CSG_depth - end_CSG_length) / length_inc)
Num_DPD_value = ((DP_depth - end_DP_length) / length_inc)

'avoid repeating addition of different end depths
If Num_WD_value = Num_DPD_value Then
end_length = end_length - end_DP_length
End If
If Num_WD_value = Num_CSD_value Then
end_length = end_length - end_CSG_length
End If
If Num_CSD_value = Num_DPD_value Then
If end_CSG_length > end_DP_length Then
end_CSG_length = end_CSG_length - end_DP_length
Else
end_DP_length = end_DP_length - end_CSG_length
End If
End If

Dim i, k As Integer
k = 1

'initialize depth and geometery array for casing depth less than drill pipe depth
If Num_CSD_value < Num_DPD_value Then
For i = 2 To (Num_CSD_value + 1)
depth_array(i) = length_inc * k
hole_diameter(i) = ID_casing
tubing_OD(i) = OD_DP
tubing_ID(i) = ID_DP
k = k + 1
Next i
depth_array(Num_CSD_value + 2) = depth_array(Num_CSD_value + 1) + end_CSG_length
hole_diameter(Num_CSD_value + 2) = ID_casing
tubing_OD(Num_CSD_value + 2) = OD_DP
tubing_ID(Num_CSD_value + 2) = ID_DP
```

```
For i = (Num_CSD_value + 3) To (Num_DPD_value + 2)
depth_array(i) = length_inc * k
hole_diameter(i) = OH_diameter
tubing_OD(i) = OD_DP
tubing_ID(i) = ID_DP
k = k + 1
Next i
depth_array(Num_DPD_value + 3) = depth_array(Num_DPD_value + 2) + end_DP_length
hole_diameter(Num_DPD_value + 3) = OH_diameter
tubing_OD(Num_DPD_value + 3) = OD_DP
tubing_ID(Num_DPD_value + 3) = ID_DP

'adjust for 0 drill collar length or length less than drill collar length
If DC_length <> 0 Then
If DC_length < length_inc Then
depth_array(Num_WD_value + 4) = depth_array(Num_WD_value + 3) + DC_length
hole_diameter(Num_WD_value + 4) = OH_diameter
tubing_OD(Num_WD_value + 4) = OD_DC
tubing_ID(Num_WD_value + 4) = ID_DC

End If
If DC_length > length_inc Then

For i = (Num_DPD_value + 4) To (Num_WD_value + 3)
depth_array(i) = length_inc * k
hole_diameter(i) = OH_diameter
tubing_OD(i) = OD_DC
tubing_ID(i) = ID_DC
k = k + 1
Next i
depth_array(Num_WD_value + 4) = depth_array(Num_WD_value + 3) + end_length
hole_diameter(Num_WD_value + 4) = OH_diameter
tubing_OD(Num_WD_value + 4) = OD_DC
tubing_ID(Num_WD_value + 4) = ID_DC

End If

Else
depth_array(Num_WD_value + 4) = depth_array(Num_WD_value + 3)
hole_diameter(Num_WD_value + 4) = OH_diameter
tubing_OD(Num_WD_value + 4) = OD_DP
tubing_ID(Num_WD_value + 4) = ID_DP
End If

End If


'initialize depth and geometery array for drill pipe depth less than casing depth
If Num_DPD_value < Num_CSD_value Then
For i = 2 To (Num_DPD_value + 1)
depth_array(i) = length_inc * k
hole_diameter(i) = ID_casing
tubing_OD(i) = OD_DP
```

```
tubing_ID(i) = ID_DP

k = k + 1
Next i
depth_array(Num_DPD_value + 2) = depth_array(Num_DPD_value + 1) + end_DP_length
hole_diameter(Num_DPD_value + 2) = ID_casing
tubing_OD(Num_DPD_value + 2) = OD_DP
tubing_ID(Num_DPD_value + 2) = ID_DP

For i = (Num_DPD_value + 3) To (Num_CSD_value + 2)
depth_array(i) = length_inc * k
hole_diameter(i) = ID_casing
tubing_OD(i) = OD_DC
tubing_ID(i) = ID_DC
k = k + 1
Next i
depth_array(Num_CSD_value + 3) = depth_array(Num_CSD_value + 2) + end_CSG_length
hole_diameter(Num_CSD_value + 3) = ID_casing
tubing_OD(Num_CSD_value + 3) = OD_DC
tubing_ID(Num_CSD_value + 3) = ID_DC

For i = (Num_CSD_value + 4) To (Num_WD_value + 3)
depth_array(i) = length_inc * k
hole_diameter(i) = OH_diameter
tubing_OD(i) = OD_DC
tubing_ID(i) = ID_DC

k = k + 1
Next i
depth_array(Num_WD_value + 4) = depth_array(Num_WD_value + 3) + end_length
hole_diameter(Num_WD_value + 4) = OH_diameter
tubing_OD(Num_WD_value + 4) = OD_DC
tubing_ID(Num_WD_value + 4) = ID_DC

End If

'initialize depth and geometery array for drill pipe and casing depth in the same length increment
If Num_DPD_value = Num_CSD_value Then

For i = 1 To (Num_DPD_value + 1)
depth_array(i) = length_inc * k
hole_diameter(i) = ID_casing
tubing_OD(i) = OD_DP
tubing_ID(i) = ID_DP

k = k + 1
Next i

If DP_depth < CSG_depth Then
depth_array(Num_DPD_value + 2) = depth_array(Num_DPD_value + 1) + end_DP_length
hole_diameter(Num_DPD_value + 2) = ID_casing
tubing_OD(Num_DPD_value + 2) = OD_DP
tubing_ID(Num_DPD_value + 2) = ID_DP
```

```
depth_array(Num_CSD_value + 3) = depth_array(Num_CSD_value + 2) + end_CSG_length
hole_diameter(Num_CSD_value + 3) = ID_casing
tubing_OD(Num_CSD_value + 3) = OD_DC
tubing_ID(Num_CSD_value + 3) = ID_DC

Else
depth_array(Num_CSD_value + 2) = depth_array(Num_CSD_value + 1) + end_CSG_length
hole_diameter(Num_CSD_value + 2) = ID_casing
tubing_OD(Num_CSD_value + 2) = OD_DP
tubing_ID(Num_CSD_value + 2) = ID_DP

depth_array(Num_DPD_value + 3) = depth_array(Num_DPD_value + 2) + end_DP_length
hole_diameter(Num_DPD_value + 3) = OH_diameter
tubing_OD(Num_DPD_value + 3) = OD_DP
tubing_ID(Num_DPD_value + 3) = ID_DP

End If

'adjust for 0 drill collar length or length less than drill collar length
If DC_length <> 0 Then
If DC_length < length_inc Then
depth_array(Num_WD_value + 4) = depth_array(Num_WD_value + 3) + DC_length
hole_diameter(Num_WD_value + 4) = OH_diameter
tubing_OD(Num_WD_value + 4) = OD_DC
tubing_ID(Num_WD_value + 4) = ID_DC


End If
If DC_length > length_inc Then
For i = (Num_DPD_value + 4) To (Num_WD_value + 3)
depth_array(i) = length_inc * k
hole_diameter(i) = OH_diameter
tubing_OD(i) = OD_DC
tubing_ID(i) = ID_DC

k = k + 1
Next i
depth_array(Num_WD_value + 4) = depth_array(Num_WD_value + 3) + end_length
hole_diameter(Num_WD_value + 4) = OH_diameter
tubing_OD(Num_WD_value + 4) = OD_DC
tubing_ID(Num_WD_value + 4) = ID_DC

End If
Else
depth_array(Num_WD_value + 4) = depth_array(Num_WD_value + 3)
hole_diameter(Num_WD_value + 4) = OH_diameter
tubing_OD(Num_WD_value + 4) = OD_DP
tubing_ID(Num_WD_value + 4) = ID_DP

End If

End If
```

```
For i = 1 To Num_WD_value + 4
Sheets("MC-Data").Cells(i + 2, 1) = depth_array(i)
Sheets("MC-Data").Cells(i + 2, 2) = hole_diameter(i)
Sheets("MC-Data").Cells(i + 2, 3) = tubing_OD(i)
Sheets("MC-Data").Cells(i + 2, 4) = tubing_ID(i)

Next i


End Sub

Sub Temp_calc(total_length, tubing_diameter, tubing_OD, hole_diameter, mud_flow_rate,
inlet_mud_temp, visc_tubing, visc_annulus, mud_k, mud_spec_heat, mud_density, formation_k,
formation_spec_heat, formation_density, surf_earth_temp, geo_grad, time, Z, temp_ann,
temp_tub, i, Tei)

Dim l, r_t, r_to, r_a, ke, kf, kt, Tti, Tes, cfl As Double

l = total_length
r_t = (tubing_diameter / 24)
r_a = hole_diameter / 24
r_to = (tubing_OD / 24)

ke = formation_k
kf = mud_k
kt = 26
Tti = inlet_mud_temp
Tes = surf_earth_temp
cfl = mud_spec_heat

Dim w, Ut, Ua As Double
'calculate w in lb/hr making the necessary conversions
w = mud_flow_rate * mud_density * 60

Ut = 27.61
Ua = 36.393

Dim alph1, td, Ttd, B, Lr As Double
'td = ke * time / (formation_spec_heat * formation_density * r_a ^ 2)
alph1 = 0.039
td = alph1 * time / (r_a ^ 2)

Ttd = Log(Exp(-0.2 * td) + (1.5 - (0.3719 * Exp(-td))) * (td ^ 0.5))
B = (w * cfl) / (2 * 3.14159 * r_t * Ut)
Lr = (2 * 3.14159 / (cfl * w)) * ((r_a * Ua * ke) / (ke + (r_a * Ua * Ttd)))

Dim lambda1, lambda2, alpha, beta, T_t, T_a As Double
lambda1 = (Lr / 2) + (Lr / 2) * ((1 + 4 * (r_a * Ua * Ttd + ke) * ((r_t * Ut) / (r_a * Ua * ke))) ^ 0.5)
lambda2 = (Lr / 2) - (Lr / 2) * ((1 + 4 * (r_a * Ua * Ttd + ke) * ((r_t * Ut) / (r_a * Ua * ke))) ^ 0.5)

alpha = -((Tti - Tes + (B * geo_grad)) * (lambda2 * Exp(lambda2 * l)) + geo_grad) / ((lambda1 *
Exp(lambda1 * l)) - (lambda2 * Exp(lambda2 * l)))
```

```
beta = ((Tti - Tes + (B * geo_grad)) * (lambda1 * Exp(lambda1 * l)) + geo_grad) / ((lambda1 * Exp(lambda1 * l)) - (lambda2 * Exp(lambda2 * l)))

T_t = (alpha * Exp(lambda1 * Z)) + (beta * Exp(lambda2 * Z)) + (geo_grad * Z) - (B * geo_grad) + Tes
T_a = ((1 + (B * lambda1)) * (alpha * Exp(lambda1 * Z))) + ((1 + (B * lambda2)) * (beta * Exp(lambda2 * Z))) + (geo_grad * Z) + Tes

temp_tub = T_t
temp_ann = T_a

Tei = (geo_grad * Z) + Tes

'display the fluid propertise on the Fluid Properties HK worksheet
Sheets("Temp Data-MC").Cells(i + 2, 1) = Z
Sheets("Temp Data-MC").Cells(i + 2, 2) = w
Sheets("Temp Data-MC").Cells(i + 2, 3) = Ut
Sheets("Temp Data-MC").Cells(i + 2, 4) = Ua
Sheets("Temp Data-MC").Cells(i + 2, 5) = td
Sheets("Temp Data-MC").Cells(i + 2, 6) = Ttd
Sheets("Temp Data-MC").Cells(i + 2, 7) = B
Sheets("Temp Data-MC").Cells(i + 2, 8) = Lr
Sheets("Temp Data-MC").Cells(i + 2, 9) = lambda1
Sheets("Temp Data-MC").Cells(i + 2, 10) = lambda2
Sheets("Temp Data-MC").Cells(i + 2, 11) = alpha
Sheets("Temp Data-MC").Cells(i + 2, 12) = beta
Sheets("Temp Data-MC").Cells(i + 2, 13) = Tei
Sheets("Temp Data-MC").Cells(i + 2, 14) = T_t
Sheets("Temp Data-MC").Cells(i + 2, 15) = T_a

End Sub

Public Static Function Log10(x)
Log10 = Log(x) / Log(10#)
End Function

Private Sub CommandButton2_Click()
'instructions to pop up
MsgBox _
 "Welcome to Gas Kick Mechanistic Model!" & vbCrLf & vbCrLf & _
 "This simulator provides estimation of the change in wellbore flow" & vbCrLf & _
 "by estimating gas flow into the wellbore and its effect on the " & vbCrLf & _
 "annulus pressure, temperature and fluid property profile" & vbCrLf & vbCrLf & _
 "Enter in the appropriate well bore data to create a mud circulation profile" & vbCrLf & _
 "and then click the 'Calculate' button.  Then proceed to 'MC-Profiles' sheet " & vbCrLf & _
 "to view the profile of the wellbore during mud circulation " & vbCrLf & vbCrLf & _
 "The proceed to the 'Gas Kick' sheet and enter appropriate reservoir and " & vbCrLf & _
 "depth of entry of gas into the wellbore. " & vbCrLf & vbCrLf & _
 "The steady state profiles will be displayed in 'After GK - SS Profiles'. " & vbCrLf & _
 "The simulated values for the profile as function of time will be displayed " & vbCrLf & _
 "in 'Timestep-AP' sheet and charts in 'Function of time' sheet " & vbCrLf & _
 " "
End Sub
```

The VBA code for simulation of gas kick into the wellbore:

```
Private Sub CommandButton1_Click()

'clear the pressure/ depth columns and the Fluid Properties Hom sheet
Sheets("After GK - SS Profiles").Cells.Range("A7:A1000").ClearContents
Sheets("After GK - SS Profiles").Cells.Range("B7:B1000").ClearContents
Sheets("After GK - SS Profiles").Cells.Range("C7:C1000").ClearContents
Sheets("After GK - SS Profiles").Cells.Range("D7:J1000").ClearContents
Sheets("After GK - SS Profiles").Cells.Range("E7:K1000").ClearContents
Sheets("After GK - SS Profiles").Cells.Range("F7:L1000").ClearContents
Sheets("After GK - SS Profiles").Cells.Range("G7:M1000").ClearContents
Sheets("After GK - SS Profiles").Cells.Range("H7:N1000").ClearContents

Sheets("Temp Data-2P").Cells.Range("A3:O1000").ClearContents
Sheets("2P-Data").Cells.Range("A3:Z1000").ClearContents
Sheets("Timestep-AP").Cells.Range("C3:EE5").ClearContents
Sheets("Timestep-AP").Cells.Range("A7:EE1000").ClearContents

Dim well_depth, CSG_depth, ID_casing, OH_diameter, DC_length, OD_DC, ID_DC, OD_DP,
ID_DP, pipe_roughness As Double
well_depth = Sheets("Mud Circulation").Cells.Range("E6").Value
CSG_depth = Sheets("Mud Circulation").Cells.Range("E7").Value
ID_casing = Sheets("Mud Circulation").Cells.Range("E8").Value
OH_diameter = Sheets("Mud Circulation").Cells.Range("E9").Value
DC_length = Sheets("Mud Circulation").Cells.Range("E10").Value
OD_DC = Sheets("Mud Circulation").Cells.Range("E11").Value
ID_DC = Sheets("Mud Circulation").Cells.Range("E12").Value
OD_DP = Sheets("Mud Circulation").Cells.Range("E13").Value
ID_DP = Sheets("Mud Circulation").Cells.Range("E14").Value
pipe_roughness = Sheets("Mud Circulation").Cells.Range("E15").Value

Dim length_inc As Double
length_inc = Sheets("Mud Circulation").Cells.Range("E17").Value

Dim mud_flow_rate, mud_density As Double
mud_flow_rate = Sheets("Mud Circulation").Cells.Range("E20").Value
mud_density = Sheets("Mud Circulation").Cells.Range("E21").Value

Dim R3, R100, R300, R600 As Double
R3 = Sheets("Mud Circulation").Cells.Range("H6").Value
R100 = Sheets("Mud Circulation").Cells.Range("H7").Value
R300 = Sheets("Mud Circulation").Cells.Range("H8").Value
R600 = Sheets("Mud Circulation").Cells.Range("H9").Value

'collect gas influx and reservoir information
Dim entry_depth, reservoir_press, perm, reservoir_ht, re, skin, gasgrav As Double
entry_depth = Range("G6").Value
reservoir_press = Range("G7").Value
perm = Range("G8").Value
reservoir_ht = Range("G9").Value
re = Range("G10").Value
```

```
skin = Range("G11").Value
gasgrav = Range("G13").Value


'check to see any invalid inputs
If entry_depth > well_depth Or entry_depth < 0 Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If

If reservoir_ht > well_depth Or reservoir_ht < 0 Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If


If Not IsNumeric(entry_depth) Or Not IsNumeric(reservoir_press) Or Not IsNumeric(perm) Or Not IsNumeric(reservoir_ht) Or Not IsNumeric(re) Or Not IsNumeric(skin) Or Not IsNumeric(gasgrav) Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If



'collect values from the excel sheet for temperature calculations
Dim inlet_mud_temp, mud_k, mud_spec_heat As Double
Dim formation_k, formation_spec_heat, formation_density As Double
Dim surf_earth_temp, geo_grad, time As Double
inlet_mud_temp = Sheets("Mud Circulation").Cells.Range("H12").Value
mud_k = Sheets("Mud Circulation").Cells.Range("H13").Value
mud_spec_heat = Sheets("Mud Circulation").Cells.Range("H14").Value
formation_k = Sheets("Mud Circulation").Cells.Range("H15").Value
formation_spec_heat = Sheets("Mud Circulation").Cells.Range("H16").Value
formation_density = Sheets("Mud Circulation").Cells.Range("H17").Value
surf_earth_temp = Sheets("Mud Circulation").Cells.Range("H18").Value
geo_grad = Sheets("Mud Circulation").Cells.Range("H19").Value
time = Sheets("Mud Circulation").Cells.Range("H20").Value

'initialize array length
Dim end_length As Double
Dim Num_WD_value As Integer
end_length = well_depth Mod length_inc
Num_WD_value = ((well_depth - end_length) / length_inc)
Array_length = Num_WD_value + 5

'calculate hole_diameter, tubing_ID, tubing_OD through the wellbore
Dim hole_diameter(), tubing_ID(), tubing_OD(), depth_array() As Double
ReDim depth_array(1 To Array_length)
ReDim hole_diameter(1 To Array_length)
ReDim tubing_ID(1 To Array_length)
ReDim tubing_OD(1 To Array_length)
```

```
''''''''''''''''''''
Dim press_tub_array(), press_ann_array, temp_tub_array(), temp_ann_array(), temp_ann_sp_array(), press_ann_sp_array(), visc_tubing(), visc_annulus() As Double
ReDim press_tub_array(1 To Array_length)
ReDim press_ann_array(1 To Array_length)
ReDim temp_tub_array(1 To Array_length)
ReDim temp_ann_array(1 To Array_length)
ReDim temp_ann_sp_array(1 To Array_length)
ReDim press_ann_sp_array(1 To Array_length)
ReDim visc_tubing(1 To Array_length)
ReDim visc_annulus(1 To Array_length)

Dim entry_index As Integer

Call Initialize_Array(entry_depth, Array_length, length_inc, depth_array, hole_diameter, tubing_OD, tubing_ID, visc_tubing, visc_annulus, press_tub_array, press_ann_array, press_ann_sp_array, temp_tub_array, temp_ann_array, temp_ann_sp_array, entry_index)

Dim gas_flow, z_gas, density_gas, pressure, temp_gas, visc_gas As Double
pressure = press_ann_sp_array(entry_index)
temp_gas = temp_ann_sp_array(entry_index)

If reservoir_press < pressure Then
MsgBox "Invalid Input" & vbCrLf
Exit Sub
End If


Call Calc_Z(gasgrav, pressure, temp_gas, z_gas)
Call Calc_gas_density_visc(gasgrav, pressure, temp, z_gas, density_gas, visc_gas)
gas_flow = (perm * reservoir_ht * (reservoir_press ^ 2 - pressure ^ 2)) / (1424 * visc_gas * z_gas * (temp_gas + 460) * (Log(re / (hole_diameter(entry_index) / 24)) + skin))

Sheets("After GK - SS Profiles").Cells(3, 2) = (gas_flow * 1000) / 1440

'initialize variable for two-phase calculation
Dim n2, k2, V2, N_reg2, f2, dpF_dl_annulus, dp_dl_hydrostatic As Double

Dim density_liquid, density_mixture, x, diameter, annulus_density As Double
Dim surface_tension, fg, Co, v00, v00T_dev, v00_avg, vgb, vann, vsga, LHS, RHS As Double
Dim fric_delp_per_foot, static_delp_per_foot, delp_per_foot, Re_m, f_m As Double
Dim Area, delp_new, delp_lower, delp_upper As Double
Dim flow_regime As String
Dim temp_mud, temp_addition As Double
Dim act_length_inc, act_length_inc2 As Double

flow_regime = "Single phase"
annulus_density = mud_density
density_mixture = mud_density * 7.4805
temp_addition = 0

Dim vg, time_step, display, j, k, total_time, total_vol, init_vol, inc_vol, display_ctr As Double
total_time = 0
```

```
total_vol = 0
display = 0
j = 0
k = 0
display_ctr = Sheets("MC-Data").Cells(1, 27)


'print initial and bottom data
For k = Array_length - 1 To 1 Step -1
Sheets("Timestep-AP").Cells(k + 6, display + 2) = press_ann_array(k)
Next k
Sheets("After GK - SS Profiles").Cells(display_ctr + 7, 1) = depth_array(Array_length)
Sheets("After GK - SS Profiles").Cells(display_ctr + 7, 2) = press_ann_sp_array(Array_length)
Sheets("After GK - SS Profiles").Cells(display_ctr + 7, 3) = temp_ann_sp_array(Array_length)
Sheets("After GK - SS Profiles").Cells(display_ctr + 7, 4) = press_ann_array(Array_length)
Sheets("After GK - SS Profiles").Cells(display_ctr + 7, 5) = temp_ann_array(Array_length)
Sheets("After GK - SS Profiles").Cells(display_ctr + 7, 6) = flow_regime
Call single_phase_pressure_loss(R3, R100, R300, R600, hole_diameter(Array_length),
tubing_OD(Array_length), n2, k2, V2, visc_annulus(Array_length), N_reg2, f2, mud_flow_rate,
mud_density, dpF_dl_annulus, dp_dl_hydrostatic, pipe_roughness)
delp_per_foot = dp_dl_hydrostatic + dpF_dl_annulus
Sheets("After GK - SS Profiles").Cells(display_ctr + 7, 7) = delp_per_foot
Sheets("After GK - SS Profiles").Cells(display_ctr + 7, 8) = annulus_density


'calculate and display annulus pressure, temperature and density as function of time
For i = Array_length - 1 To 1 Step -1

act_length_inc = depth_array(i + 1) - depth_array(i)
diameter = hole_diameter(i) - tubing_OD(i)

If (i >= entry_index) Then

pressure = press_ann_array(i + 1)
Call single_phase_pressure_loss(R3, R100, R300, R600, hole_diameter(i), tubing_OD(i), n2, k2,
V2, visc_annulus(i), N_reg2, f2, mud_flow_rate, mud_density, dpF_dl_annulus,
dp_dl_hydrostatic, pipe_roughness)
delp_per_foot = dp_dl_hydrostatic + dpF_dl_annulus
delp_new = delp_per_foot * act_length_inc
press_ann_array(i) = press_ann_array(i + 1) - delp_new
End If

'for length increment contaning the gas influx
If (i = entry_index) Then

pressure = press_ann_array(i)
temp = temp_ann_array(i)
Z = depth_array(i)
temp_mud = temp_ann_sp_array(i)

Call Fluid_properties(mud_flow_rate, gas_flow, mud_density, hole_diameter(i), tubing_OD(i),
gasgrav, z_gas, visc_gas, density_gas, density_liquid, vsg, vsl, pressure, temp, vm, Area,
gas_rate, liquid_rate)
```

```
Call Hasan_Kabir(deviation, surface_tension, diameter, x, visc_mixture, density_mixture, Co, Re_m, f_m, vsg, vsl, vsga, vm, v00, v00T_dev, vgb, v00_avg, vann, LHS, RHS, fg, flow_regime, density_gas, density_liquid, visc_gas, visc_annulus(i), fric_delp_per_foot, static_delp_per_foot, delp_per_foot, pipe_roughness)

Call Temp_two_phase_addition(mud_spec_heat, density_mixture, vsl, vsg, vm, Area, temp_gas, temp_mud, density_gas, density_liquid, temp_addition, liquid_rate, gas_rate, cp_avg, gasgrav)
Call Temp_calc(well_depth, ID_DP, OD_DP, OH_diameter, inlet_mud_temp, visc_tubing(i), visc_annulus(i), mud_k, mud_spec_heat, mud_density, density_mixture, formation_k, formation_spec_heat, formation_density, surf_earth_temp, geo_grad, time, Z, temp_ann, temp_tub, i, vm, Area, w, cp_avg, temp_addition)

temp_ann_array(i) = temp_ann

End If

If i < entry_index Then

pressure = press_ann_array(i + 1)
temp = temp_ann_array(i + 1)
Z = depth_array(i)

'call the functions to calculate dp/dz
Call Fluid_properties(mud_flow_rate, gas_flow, mud_density, hole_diameter(i), tubing_OD(i), gasgrav, z_gas, visc_gas, density_gas, density_liquid, vsg, vsl, pressure, temp, vm, Area, gas_rate, liquid_rate)
Call Hasan_Kabir(deviation, surface_tension, diameter, x, visc_mixture, density_mixture, Co, Re_m, f_m, vsg, vsl, vsga, vm, v00, v00T_dev, vgb, v00_avg, vann, LHS, RHS, fg, flow_regime, density_gas, density_liquid, visc_gas, visc_annulus(i), fric_delp_per_foot, static_delp_per_foot, delp_per_foot, pipe_roughness)
Call Temp_calc(well_depth, ID_DP, OD_DP, OH_diameter, inlet_mud_temp, visc_tubing(i), visc_annulus(i), mud_k, mud_spec_heat, mud_density, density_mixture, formation_k, formation_spec_heat, formation_density, surf_earth_temp, geo_grad, time, Z, temp_ann, temp_tub, i, vm, Area, w, cp_avg, temp_addition)

'Call Temp_calc(well_depth, tubing_ID(i), tubing_OD(i), hole_diameter(i), inlet_mud_temp, visc_tubing(i), visc_annulus(i), mud_k, mud_spec_heat, mud_density, density_mixture, formation_k, formation_spec_heat, formation_density, surf_earth_temp, geo_grad, time, Z, temp_ann, temp_tub, i, vm, Area, w, cp_avg, temp_addition)

'calculate the pressure
delp_new = delp_per_foot * act_length_inc
press_ann_array(i) = press_ann_array(i + 1) - delp_new

'temperature
temp_ann_array(i) = temp_ann

annulus_density = density_mixture / 7.48052

vg = vsg / fg
timestep = act_length_inc / vg

inc_vol = (fg * act_length_inc * Area) / 4.211
```

```
total_vol = total_vol + inc_vol

For j = (i - 1) To 1 Step -1

        act_length_inc2 = depth_array(j + 1) - depth_array(j)
        Call single_phase_pressure_loss(R3, R100, R300, R600, hole_diameter(j), tubing_OD(j),
n2, k2, V2, visc_annulus(j), N_reg2, f2, mud_flow_rate, mud_density, dpF_dl_annulus,
dp_dl_hydrostatic, pipe_roughness)
        delp_per_foot = dp_dl_hydrostatic + dpF_dl_annulus
        delp_new = delp_per_foot * act_length_inc2
        press_ann_array(j) = press_ann_array(j + 1) - delp_new

    Next j

display = display + 1
total_time = total_time + timestep

Sheets("2P-Data").Cells(i + 2, 3) = vsl
Sheets("2P-Data").Cells(i + 2, 4) = vsg
Sheets("2P-Data").Cells(i + 2, 5) = fric_delp_per_foot
Sheets("2P-Data").Cells(i + 2, 6) = static_delp_per_foot
Sheets("2P-Data").Cells(i + 2, 7) = z_gas
Sheets("2P-Data").Cells(i + 2, 8) = visc_gas
Sheets("2P-Data").Cells(i + 2, 9) = pressure
Sheets("2P-Data").Cells(i + 2, 10) = temp
Sheets("2P-Data").Cells(i + 2, 11) = v00
Sheets("2P-Data").Cells(i + 2, 12) = v00T_dev
Sheets("2P-Data").Cells(i + 2, 13) = density_gas
Sheets("2P-Data").Cells(i + 2, 14) = density_liquid
Sheets("2P-Data").Cells(i + 2, 15) = fg
Sheets("2P-Data").Cells(i + 2, 16) = f_m

End If

Sheets("Timestep-AP").Cells(i + 6, 1) = depth_array(i)
If display > 0 Then

Sheets("Timestep-AP").Cells(3, display + 2) = total_time * mud_flow_rate * (0.03175 / 60)
Sheets("Timestep-AP").Cells(4, display + 2) = total_time
Sheets("Timestep-AP").Cells(5, display + 2) = total_vol

For k = Array_length - 1 To 1 Step -1
Sheets("Timestep-AP").Cells(k + 6, display + 2) = press_ann_array(k)

Next k
End If

Sheets("2P-Data").Cells(i + 2, 1) = depth_array(i)
Sheets("2P-Data").Cells(i + 2, 2) = density_mixture
Sheets("2P-Data").Cells(i + 2, 17) = dpF_dl_annulus
Sheets("2P-Data").Cells(i + 2, 18) = dp_dl_hydrostatic
```

```
If depth_array(i) Mod length_inc = 0 Or depth_array(i) = well_depth Then
If depth_array(i) <> depth_array(i + 1) Then

Sheets("After GK - SS Profiles").Cells(display_ctr + 6, 1) = depth_array(i)
Sheets("After GK - SS Profiles").Cells(display_ctr + 6, 2) = press_ann_sp_array(i)
Sheets("After GK - SS Profiles").Cells(display_ctr + 6, 3) = temp_ann_sp_array(i)

Sheets("After GK - SS Profiles").Cells(display_ctr + 6, 4) = press_ann_array(i)
Sheets("After GK - SS Profiles").Cells(display_ctr + 6, 5) = temp_ann_array(i)

Sheets("After GK - SS Profiles").Cells(display_ctr + 6, 6) = flow_regime
Sheets("After GK - SS Profiles").Cells(display_ctr + 6, 7) = delp_per_foot
Sheets("After GK - SS Profiles").Cells(display_ctr + 6, 8) = annulus_density


display_ctr = display_ctr - 1
End If
End If


Next i




End Sub

Sub Initialize_Array(entry_depth, Array_length, length_inc, depth_array, hole_diameter, tubing_OD, tubing_ID, visc_tubing, visc_annulus, press_tub_array, press_ann_array, press_ann_sp_array, temp_tub_array, temp_ann_array, temp_ann_sp_array, entry_index)

Dim i As Integer
Dim diff As Double

'initialize arrays to be used incorporating the entry depth
For i = 1 To (Array_length - 1)

If Sheets("MC-Data").Cells(i + 2, 1) < entry_depth Then

depth_array(i) = Sheets("MC-Data").Cells(i + 2, 1)
hole_diameter(i) = Sheets("MC-Data").Cells(i + 2, 2)
tubing_OD(i) = Sheets("MC-Data").Cells(i + 2, 3)
tubing_ID(i) = Sheets("MC-Data").Cells(i + 2, 4)
visc_tubing(i) = Sheets("MC-Data").Cells(i + 2, 5)
visc_annulus(i) = Sheets("MC-Data").Cells(i + 2, 10)

press_tub_array(i) = Sheets("MC-Data").Cells(i + 2, 9)
press_ann_array(i) = Sheets("MC-Data").Cells(i + 2, 14)
press_ann_sp_array(i) = Sheets("MC-Data").Cells(i + 2, 14)
temp_tub_array(i) = Sheets("Temp Data-MC").Cells(i + 2, 14)
temp_ann_array(i) = Sheets("Temp Data-MC").Cells(i + 2, 15)
temp_ann_sp_array(i) = Sheets("Temp Data-MC").Cells(i + 2, 15)
```

```
Else
diff = entry_depth - depth_array(i - 1)
If (entry_depth = Sheets("MC-Data").Cells(i + 2, 1)) Or (diff > 0 And diff < length_inc) Then
depth_array(i) = entry_depth
hole_diameter(i) = hole_diameter(i - 1)
tubing_OD(i) = tubing_OD(i - 1)
tubing_ID(i) = tubing_ID(i - 1)
visc_tubing(i) = visc_tubing(i - 1)
visc_annulus(i) = visc_annulus(i - 1)

press_tub_array(i) = press_tub_array(i - 1) + ((Sheets("MC-Data").Cells(i + 2, 9) -
press_tub_array(i - 1)) * (diff / length_inc))
press_ann_array(i) = press_ann_array(i - 1) + ((Sheets("MC-Data").Cells(i + 2, 14) -
press_ann_array(i - 1)) * (diff / length_inc))
press_ann_sp_array(i) = press_ann_sp_array(i - 1) + ((Sheets("MC-Data").Cells(i + 2, 14) -
press_ann_sp_array(i - 1)) * (diff / length_inc))
temp_tub_array(i) = temp_tub_array(i - 1) + ((Sheets("Temp Data-MC").Cells(i + 2, 14) -
temp_tub_array(i - 1)) * (diff / length_inc))
temp_ann_array(i) = temp_ann_array(i - 1) + ((Sheets("Temp Data-MC").Cells(i + 2, 15) -
temp_ann_array(i - 1)) * (diff / length_inc))
temp_ann_sp_array(i) = temp_ann_sp_array(i - 1) + ((Sheets("Temp Data-MC").Cells(i + 2, 15) -
temp_ann_sp_array(i - 1)) * (diff / length_inc))

entry_index = i

depth_array(i + 1) = Sheets("MC-Data").Cells(i + 2, 1)
hole_diameter(i + 1) = Sheets("MC-Data").Cells(i + 2, 2)
tubing_OD(i + 1) = Sheets("MC-Data").Cells(i + 2, 3)
tubing_ID(i + 1) = Sheets("MC-Data").Cells(i + 2, 4)
visc_tubing(i + 1) = Sheets("MC-Data").Cells(i + 2, 5)
visc_annulus(i + 1) = Sheets("MC-Data").Cells(i + 2, 10)

press_tub_array(i + 1) = Sheets("MC-Data").Cells(i + 2, 9)
press_ann_array(i + 1) = Sheets("MC-Data").Cells(i + 2, 14)
press_ann_sp_array(i + 1) = Sheets("MC-Data").Cells(i + 2, 14)
temp_tub_array(i + 1) = Sheets("Temp Data-MC").Cells(i + 2, 14)
temp_ann_array(i + 1) = Sheets("Temp Data-MC").Cells(i + 2, 15)
temp_ann_sp_array(i + 1) = Sheets("Temp Data-MC").Cells(i + 2, 15)


Else
If Sheets("MC-Data").Cells(i + 2, 1) > entry_depth Then
depth_array(i + 1) = Sheets("MC-Data").Cells(i + 2, 1)
hole_diameter(i + 1) = Sheets("MC-Data").Cells(i + 2, 2)
tubing_OD(i + 1) = Sheets("MC-Data").Cells(i + 2, 3)
tubing_ID(i + 1) = Sheets("MC-Data").Cells(i + 2, 4)
visc_tubing(i + 1) = Sheets("MC-Data").Cells(i + 2, 5)
visc_annulus(i + 1) = Sheets("MC-Data").Cells(i + 2, 10)

press_tub_array(i + 1) = Sheets("MC-Data").Cells(i + 2, 9)
press_ann_array(i + 1) = Sheets("MC-Data").Cells(i + 2, 14)
press_ann_sp_array(i + 1) = Sheets("MC-Data").Cells(i + 2, 14)
```

```
temp_tub_array(i + 1) = Sheets("Temp Data-MC").Cells(i + 2, 14)
temp_ann_array(i + 1) = Sheets("Temp Data-MC").Cells(i + 2, 15)
temp_ann_sp_array(i + 1) = Sheets("Temp Data-MC").Cells(i + 2, 15)


End If
End If
End If

Next i

End Sub


Sub Hasan_Kabir(deviation, surface_tension, diameter, x, visc_mixture, density_mixture, Co, Re_m, f_m, vsg, vsl, vsga, vm, v00, v00T_dev, vgb, v00_avg, vann, LHS, RHS, fg, flow_regime, density_gas, density_liquid, visc_gas, visc_liquid, fric_delp_per_foot, static_delp_per_foot, delp_per_foot, pipe_roughness)

flow_regime = "N/A"
surface_tension = 28.5
deviation = 0

Dim st, inc, r_r As Double
st = surface_tension / 453.5
inc = 90 - deviation
vm = vsg + vsl
r_r = pipe_roughness / (diameter / 12)

v00 = 1.53 * ((((32.17 * st) * (density_liquid - density_gas)) / (density_liquid ^ 2)) ^ 0.25)
v00T_dev = 0.345 * ((((32.17 * st) * (density_liquid - density_gas)) / density_liquid) ^ 0.5) * ((Sin(inc / 360 * 2 * 3.14159)) ^ 0.5) * ((1 + Cos(inc / 360 * 2 * 3.14159)) ^ 1.2)
vsga = vsg / (Sin(inc / 360 * 2 * 3.14159))
vgb = ((0.429 * vsl) + (0.357 * v00)) * Sin(inc / 360 * 2 * 3.14159)
v00_avg = v00 * (1 - Exp((-0.1 * vgb) / (vsg - vgb))) + (v00T_dev * Exp((-0.1 * vgb) / (vsg - vgb)))
'v00_avg = v00 * (1 - Exp((-vgb) / vsg)) + (v00T_dev * Exp((-vgb) / vsg))
vann = 3.1 * (((32.17 * st) * (density_liquid - density_gas)) / (density_liquid ^ 2)) * ((Sin(inc / 360 * 2 * 3.14159)) ^ 0.25)

'calcule fg, gas fraction to determine flow regime
Dim fl As Double
Co = 1.2
fg = vsg / ((Co * (vsl + vsg)) + v00)

If v00T_dev > v00 Then
fg = vsg / ((Co * (vsl + vsg)) + v00_avg)
End If
fl = 1 - fg

'calculate friction factor to calculate flow regime
x = (vsg * density_gas) / ((vsg * density_gas) + (vsl * density_liquid))
visc_mixture = (visc_liquid * (1 - x)) + (visc_gas * x)
density_mixture = (fl * density_liquid) + (fg * density_gas)
```

```
Re_m = ((diameter / 12) * vm * density_mixture) / (visc_mixture * 0.000672)
'f_m = 0.32 * (Re_m ^ (-0.25))
f_m = 4 / ((4 * Log10((r_r / 3.7065) - ((5.0452 / Re_m) * Log10(((r_r ^ 1.1098) / 2.8257) + ((7.149
/ Re_m) ^ 0.8981))))) ^ 2)

'check for dispersed bubbly flow regime
LHS = (2 * vm ^ 1.2) * (f_m / (2 * (diameter / 12))) ^ 0.4 * (density_liquid / st) ^ 0.6 * ((0.4 * st) /
(32.17 * (density_liquid - density_gas))) ^ 0.5
RHS = 0.725 + 4.15 * ((vsg / vm) ^ 0.5)

'check for Annular
If fg > 0.85 And vsg > vann Then
flow_regime = "Annular"

Else
'check for bubbly flow regime
'If v00 > v00T_dev Or vsg < vgb Then
If fg < 0.25 Then
flow_regime = "Bubbly"
End If

'Else

'check for slug or churn flow
'If v00T_dev > v00 And vsg > vgb Then
If vsg > vgb Then
flow_regime = "Slug"

If vsg > (1.08 * vsl) Then
flow_regime = "Churn"
End If

'check for dispersed bubbly flow or churn flow
If LHS > RHS Then
If fg < 0.58 Then
flow_regime = "Dispersed Bubbly flow"
End If

If vsg > (1.08 * vsl) Then
flow_regime = "Churn"
End If

'End If
End If
End If
End If

If flow_regime = "Bubbly" Then
Co = 1.2
fg = vsg / ((Co * (vsl + vsg)) + v00)
v00_avg = v00
End If
```

```
If flow_regime = "Slug" Then
Co = 1.2
fg = vsg / ((Co * (vsl + vsg)) + v00_avg)
End If

If flow_regime = "Churn" Then
Co = 1.15
fg = vsg / ((Co * (vsl + vsg)) + v00_avg)
End If

If flow_regime = "Dispersed Bubbly flow" Then
Co = 1.2
fg = vsg / ((Co * (vsl + vsg)) + v00)
End If

If flow_regime = "Annular" Then
Co = 1
fg = vsg / ((Co * (vsl + vsg)) + v00)
End If

fl = 1 - fg
density_mixture = (fl * density_liquid) + (fg * density_gas)
Re_m = ((diameter / 12) * vm * density_mixture) / (visc_mixture * 0.000672)
f_m = 4 / ((4 * Log10((r_r / 3.7065) - ((5.0452 / Re_m) * Log10(((r_r ^ 1.1098) / 2.8257) + ((7.149
/ Re_m) ^ 0.8981))))) ^ 2)

'calc dp/dz
fric_delp_per_foot = (1 / 144) * ((f_m * vm ^ 2 * density_mixture) / (2 * 32.17 * (diameter / 12)))
static_delp_per_foot = (1 / 144) * density_mixture * Cos(deviation / 360 * 2 * 3.14159)
delp_per_foot = fric_delp_per_foot + static_delp_per_foot

End Sub

Sub Temp_calc(well_depth, tubing_diameter, tubing_OD, hole_diameter, inlet_mud_temp,
visc_tubing, visc_annulus, mud_k, mud_spec_heat, mud_density, density_mixture, formation_k,
formation_spec_heat, formation_density, surf_earth_temp, geo_grad, time, Z, temp_ann,
temp_tub, i, vm, Area, w, cp_avg, temp_addition)

Dim l, r_t, r_to, r_a, ke, kf, kt, Tti, Tes, Tei, cfl As Double

l = well_depth
r_t = (tubing_diameter / 24)
r_a = hole_diameter / 24
r_to = (tubing_OD / 24)

ke = formation_k
kf = mud_k
kt = 26
Tti = inlet_mud_temp
Tes = surf_earth_temp
cfl = mud_spec_heat
```

```
w = vm * density_mixture * Area * 3600

Dim Ut, Ua As Double

'constant used as per the original model
Ut = 27.61
Ua = 36.393

Dim alph1, td, Ttd, B, Lr As Double
'td = ke * time / (formation_spec_heat * formation_density * r_a ^ 2)
alph1 = 0.039
td = alph1 * time / (r_a ^ 2)

Ttd = Log(Exp(-0.2 * td) + (1.5 - (0.3719 * Exp(-td))) * (td ^ 0.5))
B = (w * cp_avg) / (2 * 3.14159 * r_t * Ut)
Lr = (2 * 3.14159 / (cp_avg * w)) * ((r_a * Ua * ke) / (ke + (r_a * Ua * Ttd)))

Dim lambda1, lambda2, alpha, beta, T_t, T_a As Double
lambda1 = (Lr / 2) + (Lr / 2) * ((1 + 4 * (r_a * Ua * Ttd + ke) * ((r_t * Ut) / (r_a * Ua * ke))) ^ 0.5)
lambda2 = (Lr / 2) - (Lr / 2) * ((1 + 4 * (r_a * Ua * Ttd + ke) * ((r_t * Ut) / (r_a * Ua * ke))) ^ 0.5)

alpha = -((Tti - Tes + (B * geo_grad)) * (lambda2 * Exp(lambda2 * l)) + geo_grad) / ((lambda1 *
Exp(lambda1 * l)) - (lambda2 * Exp(lambda2 * l)))
beta = ((Tti - Tes + (B * geo_grad)) * (lambda1 * Exp(lambda1 * l)) + geo_grad) / ((lambda1 *
Exp(lambda1 * l)) - (lambda2 * Exp(lambda2 * l)))

T_a = ((1 + (B * lambda1)) * (alpha * Exp(lambda1 * Z))) + ((1 + (B * lambda2)) * (beta *
Exp(lambda2 * Z))) + (geo_grad * Z) + Tes '+ temp_addition

temp_ann = T_a

Tei = (geo_grad * Z) + Tes

'display the fluid propertise on the Fluid Properties HK worksheet
Sheets("Temp Data-2P").Cells(i + 2, 1) = Z
Sheets("Temp Data-2P").Cells(i + 2, 2) = w
Sheets("Temp Data-2P").Cells(i + 2, 3) = Ut
Sheets("Temp Data-2P").Cells(i + 2, 4) = Ua
Sheets("Temp Data-2P").Cells(i + 2, 5) = td
Sheets("Temp Data-2P").Cells(i + 2, 6) = Ttd
Sheets("Temp Data-2P").Cells(i + 2, 7) = B
Sheets("Temp Data-2P").Cells(i + 2, 8) = Lr
Sheets("Temp Data-2P").Cells(i + 2, 9) = lambda1
Sheets("Temp Data-2P").Cells(i + 2, 10) = lambda2
Sheets("Temp Data-2P").Cells(i + 2, 11) = alpha
Sheets("Temp Data-2P").Cells(i + 2, 12) = beta
Sheets("Temp Data-2P").Cells(i + 2, 13) = cp_avg
Sheets("Temp Data-2P").Cells(i + 2, 14) = temp_addition
Sheets("Temp Data-2P").Cells(i + 2, 15) = T_a


End Sub
```

```
Sub Temp_two_phase_addition(mud_spec_heat, density_mixture, vsl, vsg, vm, Area, temp_gas, temp_mud, density_gas, density_liquid, temp_addition, liquid_rate, gas_rate, cp_avg, gasgrav)

Dim cfl, cpg As Double

cfl = mud_spec_heat

'gas specific heat
cpg = 0.78

Dim total_energy, temp_mixture As Double
total_energy = (density_liquid * cfl * temp_mud * liquid_rate) + (density_gas * cpg * temp_gas * gas_rate)
total_energy = (density_liquid * cfl * temp_mud * liquid_rate) + (temp_gas * gas_rate * cpg * 28.97 * gasgrav / 379)

cp_avg = ((density_liquid * cfl * liquid_rate) + (gas_rate * cpg * 28.97 * gasgrav / 379)) / ((density_liquid * liquid_rate) + (gas_rate * 28.97 * gasgrav / (0.73 * 520)))

temp_mixture = total_energy / (((density_liquid * liquid_rate) + (gas_rate * 28.97 * gasgrav / (0.73 * 520))) * cp_avg)

temp_addition = temp_mixture - temp_mud

End Sub

Sub single_phase_pressure_loss(R3, R100, R300, R600, hole_diameter, tubing_OD, n2, k2, V2, visc_annulus, N_reg2, f2, mud_flow_rate, mud_density, dpF_dl_annulus, dp_dl_hydrostatic, pipe_roughness)

n2 = 0.657 * Log10(R100 / R3)
k2 = (5.11 * R100) / (170.2 ^ n2)
V2 = (0.408 * mud_flow_rate) / (hole_diameter ^ 2 - tubing_OD ^ 2)
visc_annulus = (100 * k2) * ((144 * V2) / (hole_diameter - tubing_OD)) ^ (n2 - 1) * ((2 * n2 + 1) / (3 * n2)) ^ n2

Dim r_r, Re_m, f_m As Double
r_r = roughness / ((hole_diameter - tubing_OD) / 12)
Re_m = (((hole_diameter - tubing_OD) / 12) * V2 * mud_density * 7.4805) / (visc_annulus * 0.000672)
f_m = 4 / ((4 * Log10((r_r / 3.7065) - ((5.0452 / Re_m) * Log10(((r_r ^ 1.1098) / 2.8257) + ((7.149 / Re_m) ^ 0.8981))))) ^ 2)

f2 = f_m / 4

dpF_dl_annulus = (f2 * V2 ^ 2 * mud_density) / (25.81 * (hole_diameter - tubing_OD))

dp_dl_hydrostatic = 0.052 * mud_density


End Sub
```

```
Sub Fluid_properties(mud_flow_rate, gas_flow, mud_density, hole_diameter, tubing_OD, gasgrav, z_gas, visc_gas, density_gas, density_liquid, vsg, vsl, pressure, temp, vm, Area, gas_rate, liquid_rate)

Call Calc_Z(gasgrav, pressure, temp, z_gas)
'z_gas = 0.95

Call Calc_gas_density_visc(gasgrav, pressure, temp, z_gas, density_gas, visc_gas)

'calculate liquid_rate in ft^3 per second
liquid_rate = mud_flow_rate / 7.48052 / 60

'calculate liquid density in lb/ft^3
density_liquid = mud_density * 7.48052

'calculate gas rate in scf/sec
gas_rate = (gas_flow * 1000) / 86400

'calculate annulus area
Area = (3.141592654 / 4) * ((hole_diameter / 12) ^ 2 - (tubing_OD / 12) ^ 2)

'calculate vsg
vsg = (gas_rate / Area) * (14.7 / 520) * (temp + 460) * (z_gas / pressure)

'calculate vsl
vsl = (0.408 * mud_flow_rate) / (hole_diameter ^ 2 - tubing_OD ^ 2)
vsl = (liquid_rate / Area)

vm = vsl + vsg

End Sub

Sub Calc_Z(gasgrav, pressure, temp, z_gas)
''''''''''''''''''''''''''''''''''''
' calculate gas properties: Z

Dim Tpc, Ppc, Ppr, Tpr As Double

Tpc = 168 + (325 * gasgrav) - (12.5 * (gasgrav ^ 2))
Ppc = 677 + (15 * gasgrav) - (35.7 * (gasgrav ^ 2))
Tpr = (temp + 460) / Tpc
Ppr = pressure / Ppc

Dim A1, A2, A3, A4, A5, A6, A7, A8, A9, A10, A11, rowpr As Double

A1 = 0.3265
A2 = -1.07
A3 = -0.5339
A4 = 0.01569
A5 = -0.05165
A6 = 0.5475
A7 = -0.7361
A8 = 0.1844
```

```
A9 = 0.1056
A10 = 0.6134
A11 = 0.721

Dim S1, S2, S3 As Double

S1 = A1 + (A2 / Tpr) + (A3 / (Tpr ^ 3)) + (A4 / (Tpr ^ 4)) + (A5 / (Tpr ^ 5))
S2 = (A6 + (A7 / Tpr) + (A8 / (Tpr ^ 2)))
S3 = A9 * ((A7 / Tpr) + (A8 / (Tpr ^ 2)))

z_gas = 1

Dim j As Integer
   For j = 1 To 20

      rowpr = (0.27 * (Ppr / (z_gas * Tpr)))
      z_gas = 1 + (S1 * (rowpr)) + (S2 * (rowpr) ^ 2) - (S3 * ((rowpr) ^ 5)) + (A10 * (1 + (A11 *
((rowpr) ^ 2))) * (((rowpr) ^ 2) / (Tpr ^ 3)) * Exp((-A11) * ((rowpr) ^ 2)))
   Next j

End Sub

Sub Calc_gas_density_visc(gasgrav, pressure, temp, z_gas, density_gas, visc_gas)

density_gas = (28.97 * gasgrav * pressure) / (10.732 * z_gas * (temp + 460))

' calc gas viscosity
Dim aaa, bbb, ccc, Ma As Double
Ma = 28.97 * gasgrav
aaa = ((9.37 + (0.01607 * Ma)) * ((temp + 460) ^ 1.5)) / (209.2 + (19.26 * Ma) + (temp + 460))
bbb = 3.448 + (986.4 / (temp + 460)) + (0.01009 * Ma)
ccc = 2.447 - (0.2224 * bbb)
visc_gas = aaa * (10 ^ (-4)) * Exp(bbb * ((gasdensity / 62.4) ^ ccc))


End Sub

Public Static Function Log10(x)
Log10 = Log(x) / Log(10#)
End Function
```