



INSTITUTO UNIVERSITARIO AERONÁUTICO

Modelo Simple de Motores Stirling

Informe Técnico: DMA-012/12

Revisión: A

Proyecto: Fonarsec INTIHUASI

Fecha: 27/08/2012

Autor:

Ing. Gustavo Scarpin
Director Dpto. Mec. Aer



PROYECTO FONARSEC INTIHUASI

Modelo Simple de Motores Stirling

Por:

Ing. Gustavo Scarpin

RESUMEN

Dentro del proyecto de desarrollo llevado a cabo por el Consorcio Público Privado Intihuasi, en el que el IUA es parte, y patrocinado por el Fondo de Innovación Tecnológica Sectorial de Energía Solar 2010 que administra el Fondo Argentino Sectorial (FONARSEC), se desarrolló un modelo matemático que permite predecir el comportamiento de los intercambiadores de calor de un motor Stirling basado en el modelo adiabático ideal descrito en Ref. [1].

En este nuevo modelo se analizan los intercambiadores de calor (calentador, regenerador y enfriador) en lo que respecta a las transferencias de calor y caídas de presión, ya que estos efectos son los principales responsables de la pérdida de eficiencia total del motor.

Es de esperar que este nuevo análisis permita estimar en forma bastante aproximada las performances finales de un motor con ciclo Stirling para cualquier configuración propuesta.

En el Anexo B se presenta el programa de cómputo StirSimple.for el cual permite automatizar los cálculos. A modo de ejemplo se calcula las performances de un motor tipo Alfa.

Córdoba, 27 de agosto de 2012



ÍNDICE

1. INTRODUCCIÓN	3
2. DESARROLLO	3
2.1 Transferencia de Calor en el Regenerador.....	3
2.1.1 Modelo de Regenerador No Ideal.....	3
2.1.2 Evaluación de la Eficiencia del Regenerador.....	6
2.2 Transferencia de Calor en el Calentador y Enfriador	8
2.3 Pérdidas de Presión	10
2.3.1 Pérdida de Trabajo por Pérdida de Presión.....	10
2.3.2 Caída de Presión.....	11
3. RESULTADOS	15
4. CONCLUSIONES	16
5. REFERENCIAS.....	16
ANEXO A: GEOMETRÍA DE UNA MATRIZ TIPO TELA DE ALAMBRE	17
ANEXO B: MANUAL DE USUARIO – PROGRAMA STIRSIMPLE	23



1. INTRODUCCIÓN

El modelo de análisis Simple fue desarrollado por el Dr. Israel Urieli del Departamento de Ingeniería Mecánica de la Universidad de Ohio². En este trabajo se implementa dicho modelo con algunas pequeñas modificaciones, como por ejemplo, la adición de las pérdidas de presión en los conductos que unen los distintos intercambiadores de calor.

Una vez realizado el análisis adiabático ideal en un determinado motor de Stirling¹, es necesario evaluar los efectos, sobre el rendimiento del motor, de la transferencia de calor y la fricción del flujo en los tres intercambiadores de calor. Esto permitirá hacer un análisis de sensibilidad paramétrica, requerido para la optimización del diseño.

La transferencia de calor por convección forzada es fundamental para el funcionamiento del motor Stirling. El calor se transfiere desde la fuente de calor externa hacia el fluido de trabajo en la sección del calefactor, se almacena y recupera cíclicamente en el regenerador, y finalmente expulsado por el fluido de trabajo hacia el disipador de calor externo en la sección más fría. Todo esto se hace en los intercambiadores de calor compactos (gran área mojada en relación al volumen) a fin de limitar el "espacio muerto" a un valor aceptable y por lo tanto permitir una salida razonable de potencia específica del motor. Se observa que un eficaz intercambio de calor tiene como precio una alto valor de fricción en el flujo, lo que resulta en la llamada "pérdida de bombeo". Esta pérdida se refiere a la energía mecánica necesaria para "bombear" el fluido de trabajo a través de los intercambiadores de calor, y reduciendo así la potencia de salida neta del motor.

La teoría y el análisis de estos efectos son extremadamente complejos, y se llega finalmente a la conclusión que sólo se puede confiar en la gran cantidad estudios experimentales y empíricos documentados, por ejemplo los presentados en Ref. [3]. Casi toda esta vasta obra se basa en condiciones de flujo estacionario y no es, por lo tanto, directamente aplicable a las condiciones de flujo oscilantes que se observa en los motores Stirling. En este modelo se adopta un enfoque de flujo "cuasi-estacionario" en el que se supone que, en cada instante del ciclo, el flujo se comporta como si el caudal fuese constante. Este análisis fue bautizado por el Dr. Urieli como "Simple", ya que se considera que es una burda simplificación de un proceso sumamente complejo.

En la actualidad existe una gran controversia acerca de este enfoque, por lo que hay que tratar a los resultados de este análisis con una medida saludable de escepticismo. Sin embargo existen comparaciones con resultados experimentales que muestran una excelente concordancia⁴, lo cual da cierta confianza para usar el mismo en el diseño de nuevos motores. Dicho análisis se utilizará en el diseño y dimensionamiento del primer prototipo de concepto del Proyecto Inthiuasi⁵. Una vez construido el prototipo los resultados obtenidos con este modelo serán contrastados experimentalmente.

2. DESARROLLO

2.1 Transferencia de Calor en el Regenerador

2.1.1 Modelo de Regenerador No Ideal

Por definición, un regenerador es un dispositivo cíclico. En la primera parte del ciclo el gas caliente fluye a través del regenerador desde el calentador al enfriador, y al hacerlo transfiere calor a la matriz del regenerador. Esto es conocido en la literatura como "*single blow*" (soplido único). Posteriormente durante la segunda parte del ciclo, el gas frío fluye en la dirección inversa, absorbiendo el calor que se ha almacenado previamente en la matriz. Por lo tanto en el estado

estacionario la transferencia neta de calor por ciclo entre el gas de trabajo y la matriz del regenerador es cero.

La calidad del regenerador es normalmente definida en términos de la eficiencia ε del regenerador basados en la entalpía tal como sigue:

$$\varepsilon \equiv \frac{\text{cambio de entalpía del gas al atravesar el regenerador durante un soplido único}}{\text{cambio de entalpía teórica máxima equivalente en un regenerador ideal}}$$

Sin embargo, esta definición no es susceptible de utilización en los motores Stirling. Por ello se propone una definición equivalente en el contexto del modelo ideal adiabático², que representa la medida limitante del máximo rendimiento tal como sigue:

$$\varepsilon \equiv \frac{\left(\text{cantidad de calor transferido desde la matriz al gas} \right)}{\left(\text{al atravesar el regenerador durante un soplido único} \right)} \frac{\left(\text{cantidad equivalente de calor transferido en el} \right)}{\left(\text{regenerador en un modelo Adiabático Ideal} \right)}$$

La eficiencia del regenerador ε , por lo tanto varía desde 1 para un regenerador ideal (como se define en el modelo adiabático ideal¹) a 0 para ninguna acción regenerativa.

Por ejemplo, considerando el diagrama energía-theta obtenido del análisis Ideal adiabático para el motor Ross D-90 presentado en Ref. [1], se tiene:

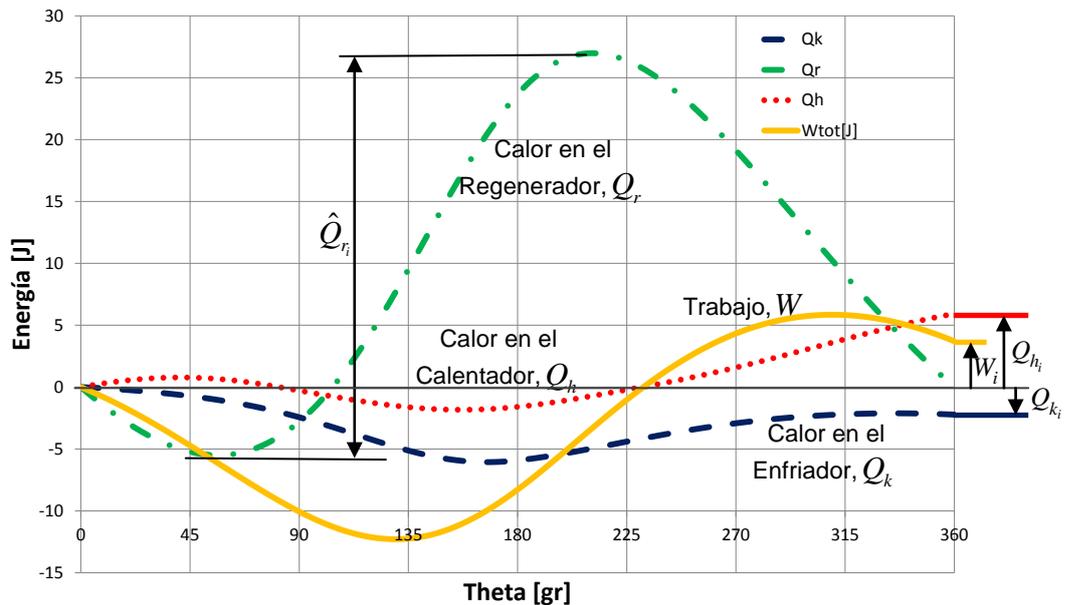


Figura 1: Diagrama Energía-Theta – Motor Ross D-90

La eficiencia térmica del ciclo Adiabático Ideal (subíndice "i") se da en términos de los valores de energía acumulada al final del ciclo a través de:

$$\eta_i = \frac{W_i}{Q_{h_i}} = \frac{Q_{h_i} + Q_{k_i}}{Q_{h_i}}$$

De la Figura 1 se puede observar que Q_{k_i} es un valor negativo, resultando para el motor Ross D-90, $\eta_i = 0.625$.

Se debe tener en cuenta también la gran cantidad de calor transferido en un soplido único del regenerador dado por \hat{Q}_{r_i} . Así, para el motor D-90, como se muestra, la relación $\hat{Q}_{r_i} / Q_{h_i} = 5.48$.

Ahora, para un sistema que tiene un regenerador no ideal, durante un soplido único, cuando el gas de trabajo fluye desde el enfriador hacia el calentador, a la salida del regenerador se tendrá una temperatura algo más baja que la del calentador. Esto dará lugar a que desde el calentador se deberá suministrar externamente más calor sobre el ciclo con el propósito de aumentar la temperatura del gas. Se puede escribir cuantitativamente como sigue:

$$Q_h = Q_{h_i} + \hat{Q}_{r_i} (1 - \varepsilon)$$

De manera similar, cuando el gas de trabajo fluye desde el calentador al refrigerador, una carga de refrigeración adicional será sumada al refrigerador, como sigue:

$$Q_k = Q_{k_i} - \hat{Q}_{r_i} (1 - \varepsilon)$$

La eficiencia térmica del motor no ideal (sin el sufijo "i") viene dada por:

$$\eta = \frac{W}{Q_h} = \frac{Q_h + Q_k}{Q_h}$$

Sustituyendo Q_h , Q_k y η_i de las ecuaciones anteriores se obtiene:

$$\eta = \frac{\eta_i}{\left[1 + \left(\frac{\hat{Q}_{r_i}}{Q_{h_i}} \right) (1 - \varepsilon) \right]}$$

En el siguiente diagrama se presenta una gráfica de la ecuación anterior para el caso específico del motor D-90 y muestra el efecto de la eficiencia del regenerador, ε , en la eficiencia térmica η .

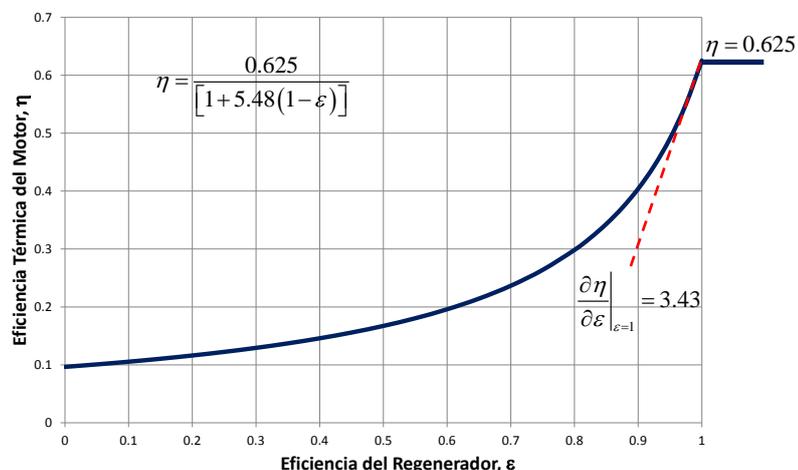


Figura 2: Diagrama Eficiencia – Motor Ross D-90

En la Figura 2 se observa que, como ε varía desde 1 para un regenerador ideal (ciclo adiabático ideal) a 0 para ninguna acción regenerativa, la eficiencia térmica η cae desde más de 60% a menos del 10%. Por otra parte, diferenciando la ecuación de eficiencia η con respecto a ε , y evaluando para $\varepsilon = 1$ se obtiene:

$$\frac{\partial \eta}{\partial \varepsilon} = \eta_i \frac{\hat{Q}_{r_i}}{Q_{h_i}}$$

Para el caso del motor Ross D-90: $\frac{\partial \eta}{\partial \varepsilon} = 0.625 \cdot 5.48 = 3.43$

Se observa que para regeneradores de gran eficiencia (cerca de $\varepsilon = 1$) una reducción del 1% en la eficiencia del regenerador implica una reducción de más de 3% en la eficiencia térmica η . Por otra parte, se ve que si se tiene un regenerador que posee una eficiencia $\varepsilon = 0,8$, el rendimiento térmico, en este caso, se reduce a la mitad, alrededor del 30%. Esto no sólo significa que se obtiene un motor mucho menos eficiente, sino que además se debe disponer de un calentador y enfriador significativamente mayores.

2.1.2 Evaluación de la Eficiencia del Regenerador

Seguidamente se analiza la eficiencia del regenerador, ε , en términos del perfil de temperatura del flujo de gas “caliente” y “frío” con respecto a de la matriz del regenerador. Se asume una diferencia de temperatura ΔT tanto para el flujo caliente como frío. Asumiendo perfiles lineales de temperatura, se llega a la definición de la eficiencia del regenerador en términos de temperatura, tal como se muestra:

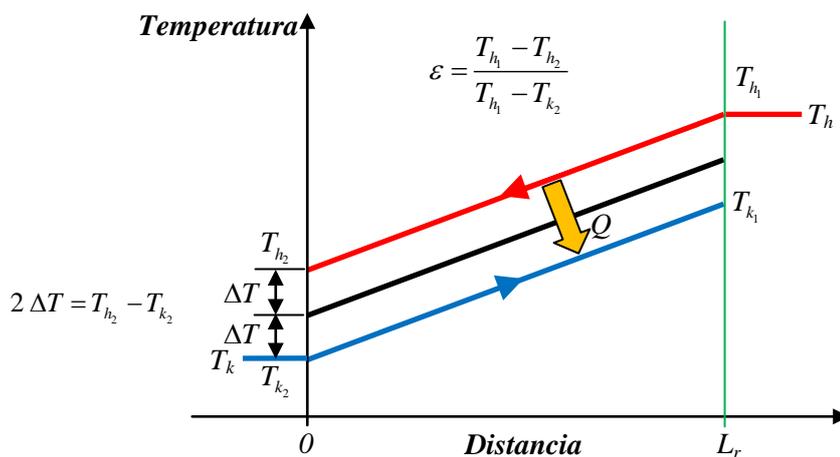


Figura 3: Perfil de temperatura en el Regenerador

Combinando las dos ecuaciones de la Figura 3, se tiene:

$$\varepsilon = \frac{1}{1 + \left(\frac{2 \Delta T}{T_{h_1} - T_{h_2}} \right)}$$



Teniendo en cuenta el balance de energía, el cambio de entalpía del flujo de gas caliente es igual a la transferencia de calor desde el gas caliente a la matriz, y posteriormente de la matriz al gas frío, por lo tanto:

$$\dot{Q} = c_p \dot{m} (T_{h_1} - T_{h_2}) = 2 h A_{wg} \Delta T$$

donde \dot{Q} (watts) es el flujo de calor, h es el coeficiente de transferencia de calor total (gas caliente→matriz→gas frío), A_{wg} es el área mojada por el gas, c_p es el calor específico a presión constante y \dot{m} (kg/s) es el flujo másico que atraviesa el regenerador. Sustituyendo la ecuación anterior en la de eficiencia del regenerador ε , se obtiene:

$$\varepsilon = \frac{1}{1 + \left(\frac{c_p \dot{m}}{h A_{wg}} \right)}$$

Introduciendo el concepto de Número de Unidades de Transferencia (NTU), muy usado en los estudios de los intercambiadores de calor, el cual es definido como:

$$NTU \equiv \frac{h A_{wg}}{c_p \dot{m}}$$

la eficiencia del regenerador se puede escribir como:

$$\varepsilon = \frac{NTU}{1 + NTU}$$

Se observa que el valor de NTU es tanto una función del tipo de intercambiador de calor, como de sus dimensiones físicas. Esto incluye el área mojada A_{wg} y el flujo de masa que atraviesa el regenerador \dot{m} .

En el análisis de los intercambiadores de calor⁶ es usual evaluar los parámetros del intercambiador en términos de las propiedades del fluido lo cual es independiente del tamaño. Para ello se hace uso del número de Stanton, el cual es definido como:

$$N_{ST} = \frac{h}{\rho u c_p}$$

donde ρ es la densidad y u es la velocidad del fluido. Por otro lado, $\dot{m} = \rho u A$, donde A es el área de flujo libre que atraviesa la matriz. Con lo anterior se puede obtener el valor de NTU en términos del número Stanton como sigue:

$$NTU = \frac{N_{ST} A_{wg}}{2 A}$$

El factor de 2 en la última ecuación es inusual, y se deriva del hecho de que el número Stanton se define generalmente por la transferencia de calor de la corriente de gas a la matriz solamente, mientras que el uso de NTU en esta oportunidad es para la transferencia global de calor del corriente caliente a la matriz regenerador, y posteriormente a la corriente fría.

En textos sobre intercambiadores de calor, existen tablas y gráficas de valores empíricos de número Stanton vs número de Reynolds para diferentes tipos de intercambiadores, por ejemplo para matrices tipo tela de alambre tejida⁶:

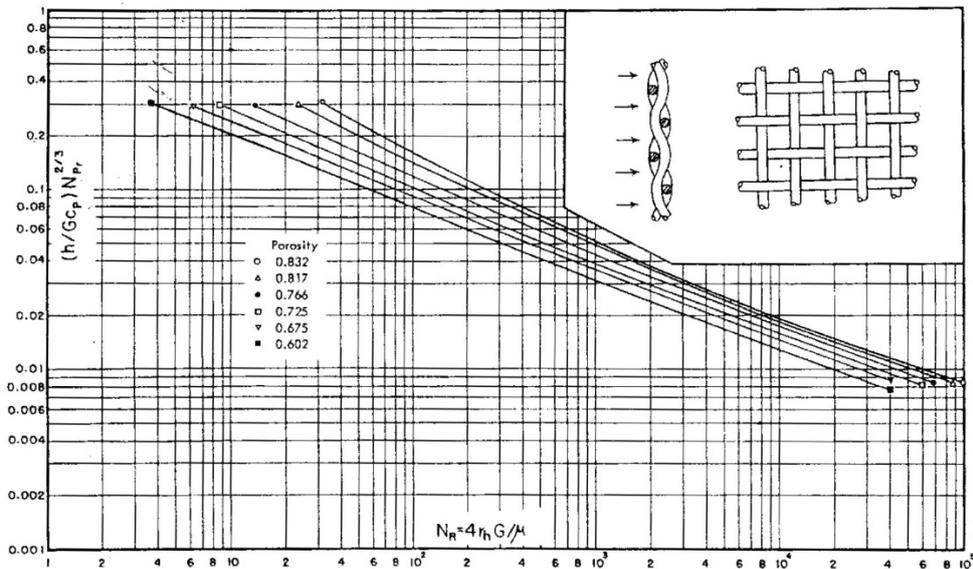


Fig. 5.7 Correlation of Stanton number, N_{st} , with Reynolds number, N_{re} , in the graphical representation pioneered by Kays and London. Reproduced with permission of McGraw-Hill Ltd and of Professors W. M. Kays and A. L. London

Figura 4: Número de Stanton vs. Número de Reynolds para matriz tipo tela de alambre⁶

Sin embargo, debido a la naturaleza inestacionaria del flujo en un motor Stirling, los valores de número de Stanton obtenidos mediante mediciones en condiciones estacionarias son poco representativos.

2.2 Transferencia de Calor en el Calentador y Enfriador

En la Figura 5 se puede observar que para un calentador y regenerador no ideales se tiene que la temperatura media efectiva del gas en la zona de calentamiento (T_h) resulta menor que en la pared del calentador (T_{w_h}). Igualmente, para un enfriador y regenerador no ideal se tiene que la temperatura media efectiva del gas en la zona de enfriamiento (T_k) resulta mayor que en la pared del enfriador (T_{w_k}). Esto tiene un efecto significativo en el rendimiento del motor, ya que está operando efectivamente con una diferencia de temperatura inferior a los existentes entre las paredes del calentador y enfriador. Por esta razón, el modelo Simple del calentador y enfriador determina iterativamente las diferencias de temperatura usando las ecuaciones de transferencia de calor por convección. Los valores de Q_h y Q_k son evaluados mediante el Modelo Adiabático Ideal¹ y el valor de la entalpía pedida por el regenerador $Q_{r_{loss}}$ en términos de la eficiencia del regenerador.

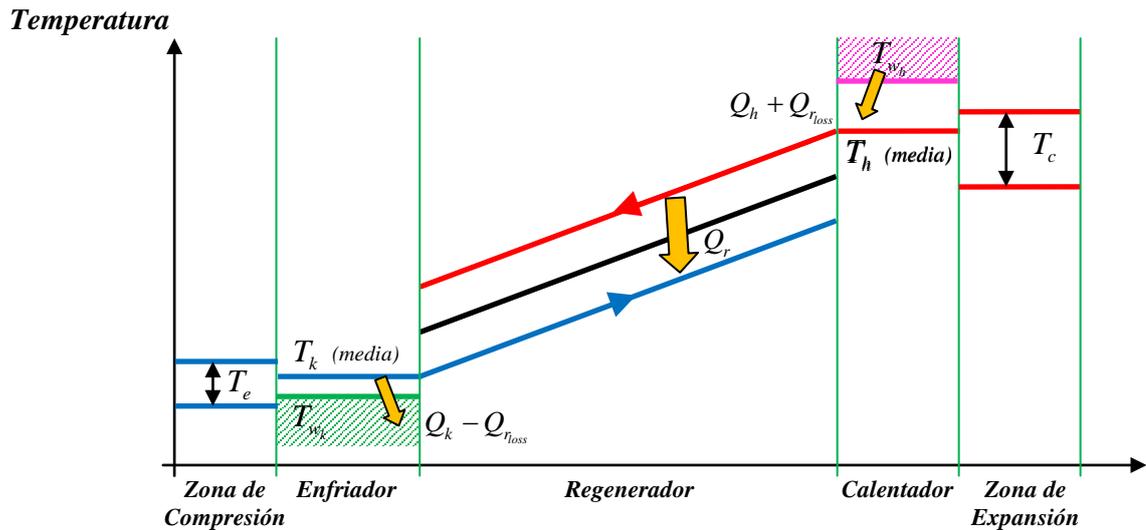


Figura 5: Distribución de Temperatura para el modelo Simple²

De la ecuación básica de transferencia de calor por convección se obtiene

$$\dot{Q} = h A_{wg} (T_w - T)$$

donde \dot{Q} (watts) es el flujo de calor total (incluida la pérdida de entalpía en el regenerador), h es el coeficiente de transferencia de calor por convección, A_{wg} es el área mojada por el gas en el intercambiador, T_w es la temperatura en la pared y T la temperatura del gas.

Con el propósito de reducir las unidades de la ecuación anterior para obtener el calor neto transferido por ciclo Q (joule/ciclo) se divide ambos miembros por la frecuencia de funcionamiento ($freq$), resultando:

$$Q_k - Q_{r_{loss}} = \frac{h_k A_{wg_k} (T_{w_k} - T_k)}{freq}$$

$$Q_h + Q_{r_{loss}} = \frac{h_h A_{wg_h} (T_{w_h} - T_h)}{freq}$$

donde, tal como se observa en la Figura 5, el subíndice h hace referencia al calentador, y k al enfriador. Seguidamente se reescriben las ecuaciones anteriores con el propósito de evaluar las temperaturas de gas T_k y T_h :

$$T_k = T_{w_k} - \frac{(Q_k - Q_{r_{loss}}) freq}{h_k A_{wg_k}}$$

$$T_h = T_{w_h} - \frac{(Q_h + Q_{r_{loss}}) freq}{h_h A_{wg_h}}$$

El modelo propuesto requiere una solución iterativa invocando a la simulación Adiabática Ideal¹ para cada nuevo valor de T_k y T_h , hasta que se alcanza la convergencia. Luego de cada simulación se dispone de los valores de Q_k y Q_h . Q_{loss} es obtenido a partir de la eficiencia del regenerador:

$$Q_{loss} = (1 - \varepsilon) \hat{Q}_r$$

El caudal másico que atraviesa el calentador y enfriador se utilizan para determinar los números de Reynolds promedio y con ello los coeficientes de transferencia de calor necesarios para evaluar las ecuaciones de T_k y T_h .

A modo de ejemplo, en la Figura 6 se muestra la distribución de temperatura obtenida de una simulación Simple del motor Ross D90. Se puede observar que la temperatura media del gas en la zona del calentador es de 59 grados por debajo de la de la pared del calentador, y de manera similar la temperatura media del gas en la zona fría es de 15 grados por encima de la de la pared del enfriador. Esta disminución en la diferencia de temperatura de funcionamiento reduce la potencia de salida de 178 W a 147 W

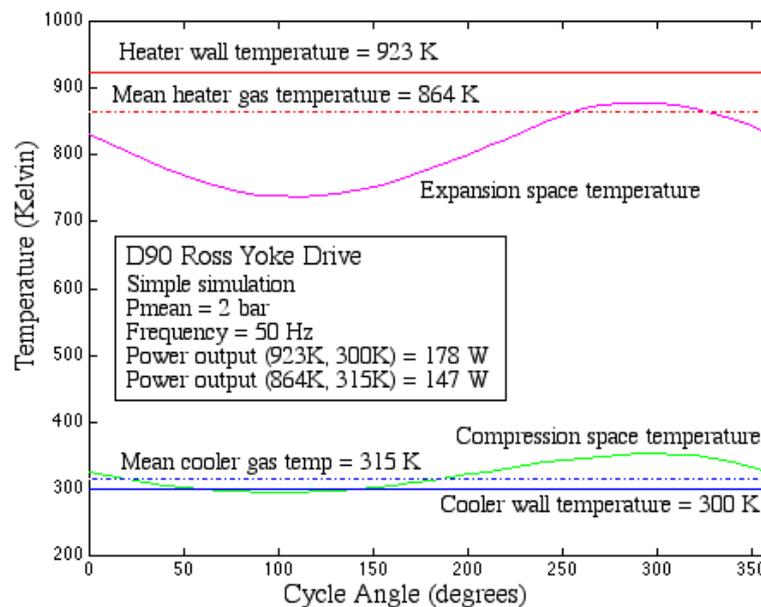


Figura 6: Modelo Simple - Diagrama Temperatura-Theta – Motor Ross D-90

2.3 Pérdidas de Presión

2.3.1 Pérdida de Trabajo por Pérdida de Presión

Hasta ahora se ha supuesto que en todo momento la presión es constante a lo largo de todo el motor. Sin embargo, el alto flujo de calor requerido en los intercambiadores de calor, a su vez, requieren que los mismos posean grandes superficies mojadas A_{wg} . Este requerimiento sumado al de mantener al mínimo los volúmenes muertos resultan en intercambiadores de calor con múltiples pasaje de fluido de pequeño diámetro y en paralelo. La fricción de fluido asociada con el flujo a través de los intercambiadores de calor producirá de hecho una caída de presión a través de cada uno de ellos que tiene el efecto de reducir la potencia de salida del motor. Esto se

conoce como la "pérdida por bombeo" y en esta sección se intentará cuantificar esta pérdida de potencia.

En primer lugar, se evaluará la caída de presión a través de los tres intercambiadores de calor con respecto a la zona de compresión. Posteriormente se determinará el nuevo valor del trabajo realizado mediante integración durante todo el ciclo, aislando el término de pérdida por bombeo.

El trabajo total durante un ciclo resulta:

$$W = W_e + W_c = \oint p dV_c + \oint (p - \sum \Delta p) dV_e$$

donde la sumatoria $\sum \Delta p$ tiene en cuenta la caída de presión en los 3 intercambiadores de calor y en los 2 tubos de conexión, por lo tanto:

$$W = \oint p (dV_c + dV_e) - \oint \sum \Delta p dV_e = W_i - \Delta W$$

donde W_i es el trabajo por ciclo obtenido del modelo Adiabático Ideal y ΔW es la pérdida de trabajo por ciclo debido a la caída de presión causado por la pérdida por bombeo. Finalmente, el término de trabajo perdido por bombeo se puede calcular como:

$$\Delta W = \int_0^{2\pi} \left(\sum_{j=1}^5 \Delta p_j \frac{dV_e}{d\theta} \right) d\theta$$

2.3.2 Caída de Presión

La caída de presión Δp es debido a la fricción del fluido a medida que fluye a través de las secciones de los intercambiadores de calor y tubos de conexión. El modelo Simple ha asumido flujo unidimensional en todas partes, sin embargo, los conceptos fundamentales de las fuerzas de fricción no pueden ser tratadas como tal. Ley de viscosidad de Newton establece que la tensión de corte τ entre las capas adyacentes de fluido es proporcional al gradiente de velocidad (du/dz) existente entre estas capas normales a la dirección de flujo, como se muestra:

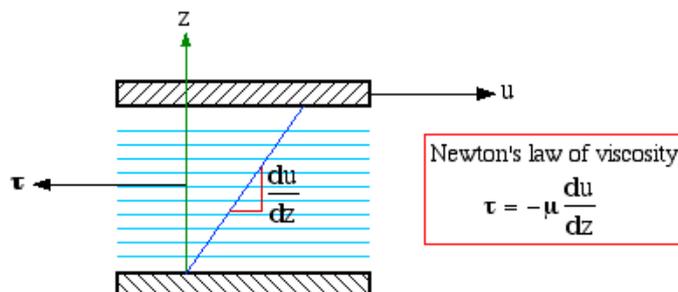


Figura 7: Ley de viscosidad de Newton

De la ecuación se observa que un fluido newtoniano no puede sostener un esfuerzo de corte a menos que el flujo sea de dos dimensiones. Esta paradoja es anulada asumiendo que el flujo no es estrictamente unidimensional, sino más bien representado por su velocidad de flujo másico medio. La viscosidad dinámica μ es básicamente una medida de la fricción interna que se produce cuando las moléculas del fluido en una capa colisionan con las moléculas en las capas adyacentes que viajan a velocidades diferentes, y, al hacerlo transfieren su impulso.

En el rango de presión de interés, la viscosidad dinámica μ es independiente de la presión. Su dependencia con temperatura para los gases de interés se obtiene del siguiente diagrama².

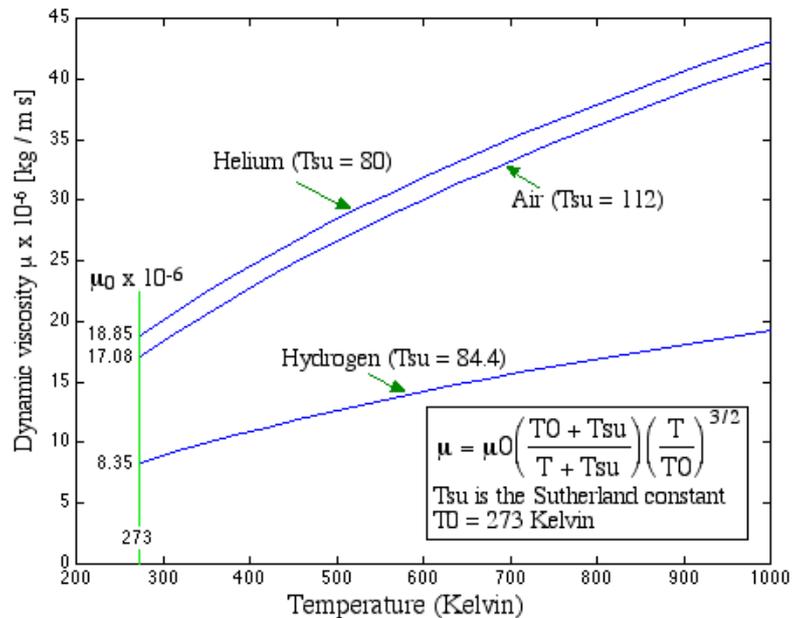


Figura 8: Formula de Sutherland para la viscosidad dinámica

La viscosidad, así como otros parámetros que define el estado de cualquier gas, para prácticamente cualquier rango de presión y temperatura, pueden ser obtenidos de Ref. [7].

La fuerza de fricción F está relacionada con la tensión de corte de la forma:

$$F = \tau A_{wg}$$

donde A_{wg} es el área mojada de la pared de los intercambiadores de calor o tubos de conexión.

Para describir la pérdida por bombeo se introduce el concepto de "diámetro hidráulico", d , el cual relaciona las dos variables más importantes de un intercambiador de calor, el volumen V vacío y el área mojada A_{wg} :

$$d = 4 \frac{V}{A_{wg}}$$

De las dos ecuaciones anteriores:

$$F = 4 \tau \frac{V}{d}$$

Definiendo el Coeficiente de Fricción como:

$$C_f \equiv \frac{\tau}{\frac{1}{2} \rho u^2}$$



donde ρ es la densidad y u es la velocidad media del fluido¹. Sustituyendo τ se puede obtener la fuerza de fricción en términos del Coeficiente de Fricción:

$$F = 2 C_f \rho u^2 \frac{V}{d}$$

Asumiendo flujo cuasi-estacionario, es decir que no existen fuerzas de inercia producto de las aceleraciones del fluido, la fuerza de fricción es igual y opuesta a la fuerza por pérdida por presión, es decir:

$$F + \Delta p A = 0$$

donde A es la sección transversal de pasaje del fluido. Sustituyendo F , la caída de presión resulta:

$$\Delta p + 2 C_f \rho u^2 \frac{V}{A d} = 0$$

Tenga en cuenta que Δp puede ser positivo o negativo, dependiendo de la dirección del flujo. Sin embargo, el segundo término de esta ecuación es siempre positivo, y por lo tanto la ecuación viola el principio de conservación de la cantidad de movimiento al invertirse el flujo. Para resolver esto se define el "Coeficiente de Fricción de Reynolds" (C_{ref}) multiplicando el número de Reynolds por el coeficiente de fricción de la siguiente manera:

$$C_{ref} = N_{Re} C_f$$

donde $N_{Re} = \rho u d / \mu$ es el Número de Reynolds, el cual por definición, es siempre positivo, independientemente de la dirección del flujo. Finalmente:

$$\Delta p = -2 C_{ref} \mu u \frac{V}{A d^2}$$

Esta ecuación satisface el principio de conservación de la cantidad de movimiento tanto para flujo positivo o reversible, ya que el signo de Δp es siempre relacionado correctamente con el signo de la velocidad u . Dado que todos los datos empíricos actuales del Coeficiente de Fricción se presentan como una función del número de Reynolds, es simple convertir dichos datos para obtener el Coeficiente de Fricción de Reynolds. Por ejemplo, durante el último medio siglo han sido de uso intensivo las curvas de Coeficiente de Fricción vs número de Reynolds para tubos circulares (Diagrama Moody). Estas curvas se han simplificado y reorganizado en términos de la Coeficiente de Fricción de Reynolds, C_{ref} , tal como se muestra en la Figura 9.

¹ $C_f = f/4$, siendo f el Factor de Fricción de Darcy según la definición del Cap. 6 de Ref. [8].

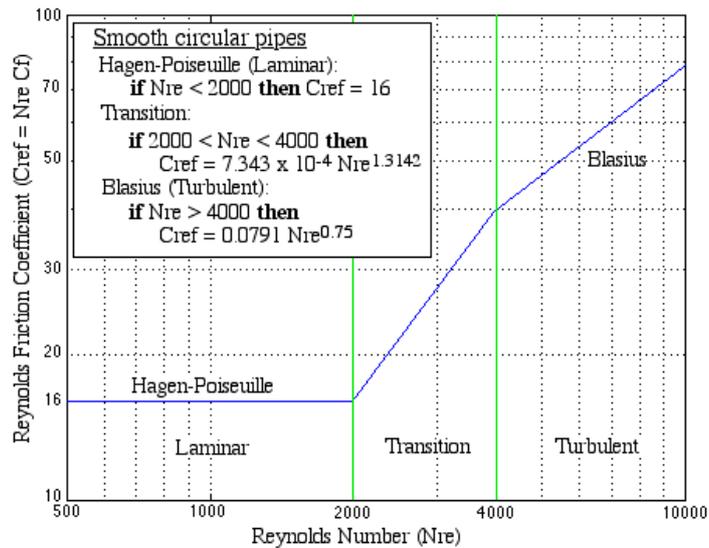


Figura 9: Diagrama de Moody en términos del Coeficiente de Fricción de Reynolds

Para intercambiadores con matriz tipo malla de alambre se puede adoptar el Coeficiente de Fricción de Reynolds sugerido por Kays & London (1955 edition)

$$C_{ref} = 54 + 1.43 N_{Re}^{0.78}$$

Más recientemente otros autores han sugerido diferentes funciones dependientes no sólo del número de Reynolds sino también de la geometría de la malla⁹.

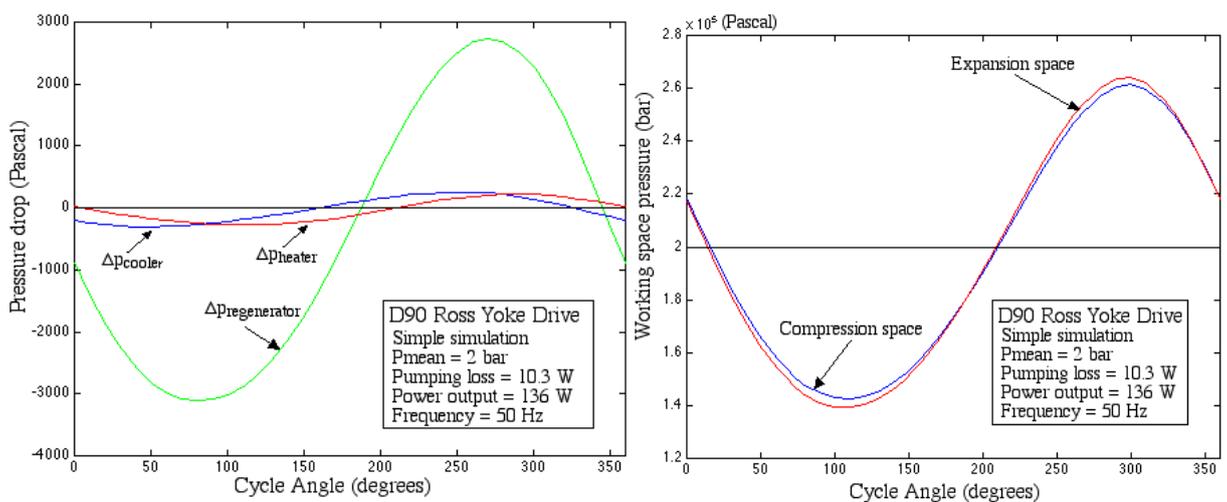


Figura 10: Modelo Simple – Efectos de las pérdida por bombeo – Motor Ross D-90²

En el ejemplo presentado en la Figura 10 se observa la gran pérdida de presión que ocurre en el regenerador respecto de los otros intercambiadores de calor. Se puede ver que la pérdida por bombeo (10.3 W) representa en este caso el 7.5% de la potencia de salida.

3. RESULTADOS

A modo de ejemplo de utilización se presenta en el Anexo B el análisis de un motor Stirling de características similares al prototipo conceptual del proyecto Intihuasi. Con este modelo se pueden obtener el mismo tipo de información que el presentado en el informe del modelo adiabático (Ref. [1]) además de la pérdida de presión obtenida con el modelo Simple.

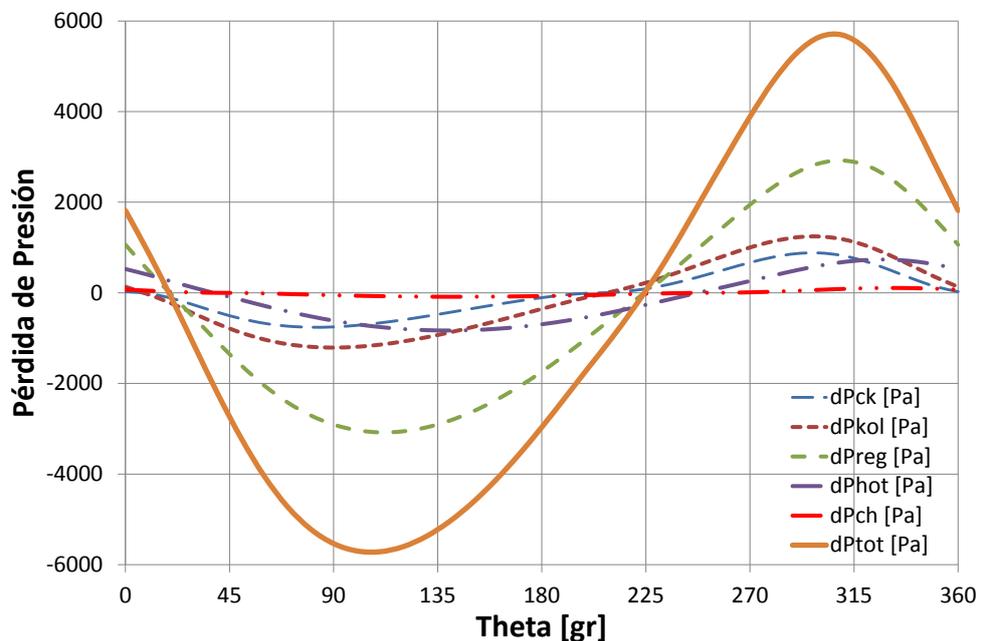


Figura 11: Modelo Simple – Efectos de las pérdidas por bombeo – (Ej. Anexo B)

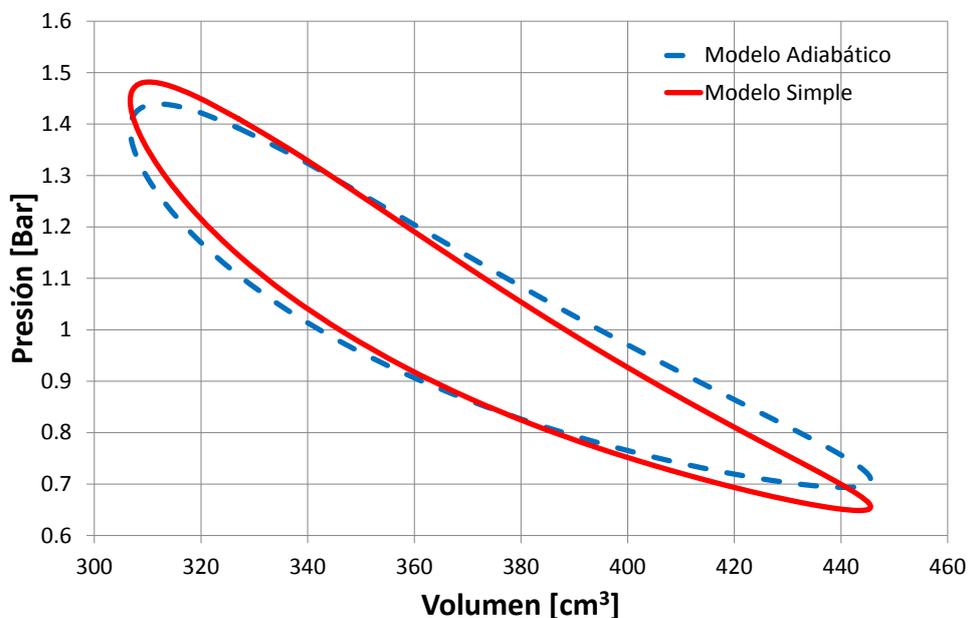


Figura 12: Modelo Simple – Gráfico Presión Volumen – (Ej. Anexo B)



4. CONCLUSIONES

En el marco del proyecto Stirling-Solar se ha desarrollado un programa de cómputos que permite predecir el comportamiento global y de cada una de las partes de un motor con ciclo Stirling.

El programa resuelve las ecuaciones de un modelo adiabático ideal de motor Stirling, incorporándose en este modelo las pérdidas de calor y de presión, lo cual resulta en una mejor aproximación al motor real.

Este tipo de programa es una herramienta importante ya que permite modificar las principales variables de diseño y rápidamente ver cómo ellas afectan al rendimiento del motor.

Es de esperar que el modelo desarrollado en este informe permita representar las performances de un motor Stirling en forma muy cercana a la realidad. Esto será verificado en ensayos que se realizarán sobre un motor real.

5. REFERENCIAS

- [1] Gustavo Scarpin, "Modelo Adiabático Ideal de Motores Stirling", Inf. DMA-003/12, IUA-FI-DMA, 28/02/2012.
- [2] Israel Urieli, "Stirling Cycle Machine Analysis - Stirling Engine Simple Analysis", Ago-2012. <http://www.ohio.edu/mechanical/stirling/simple/simple.html>
- [3] J.E. Hesselgraves, "Compact Heat Exchangers – Selection, Design and Operation", Elsevier Science & Technology Books, ISBN: 0080428398, Jan-2001.
- [4] H. Snyman, T.M. Harms y J.M. Staruss, "Design analysis methods for Stirling engines", University of Stellenbosch, Journal of Energy in Southern Africa - Vol 19 No 3, South Africa, August 2008. <http://www.erc.uct.ac.za/jesa/volume19/19-3jesa-snymanetal.pdf>
- [5] Gustavo Scarpin, "Proyecto: Fonarsec INTIHUASI - Organización del Proyecto", Inf. DMA-001/12, IUA-FI-DMA, 23/02/2012.
- [6] Theodor Finkelstein y Allan J Organ, "Air Engines - The History, Science, and Reality of the Perfect Engine", The American Society of Mechanical Engineers, New York, ISBN 0-7918-0171-3, 2001.
- [7] Gustavo Scarpin, "Cálculo de las Propiedades de Mezclas de Gases en Equilibrio Termodinámico - Aplicación A Flujos No Viscosos", Tesis de Maestría en Ciencias de la Ingeniería – Mención Aeroespacial, UNC-IUA, 2006.
- [8] Frank M. White, "Fluid Mechanics, 5th Edition", McGraw-Hill Series in Mechanical Engineering, Nov-2002. ISBN-10: 0072402172.
- [9] Richard A. Wirtz, "High Performance Woven Mesh Heat Exchange", University of Nevada-Mechanical Engineering Department, Reno, April-2003.
- [10] Telas Metálicas Casilda S.R.L., Página WEB: <http://www.telascasildasrl.com.ar/>, Set-2010.
- [11] Alan J. Organ, " 'Natural' Coordinates for Analysis of the Practical Stirling Cycle", Engineering Department, University of Cambridge, 1992.

ANEXO A: GEOMETRÍA DE UNA MATRIZ TIPO TELA DE ALAMBRE**A-1. Telas de Alambre Comerciales**

Generalmente desde el punto de vista comercial¹⁰ la geometría de una tela o malla de alambre es definida por:

Malla: En la tela metálica la malla se señala con un número que indica la cantidad de aberturas (espacios entre los alambres) que entran en una pulgada lineal (25,4 mm). Para conocer el N° de mallas, se cuentan, empezando desde el eje de un alambre, la cantidad de espacios que hay en una pulgada.

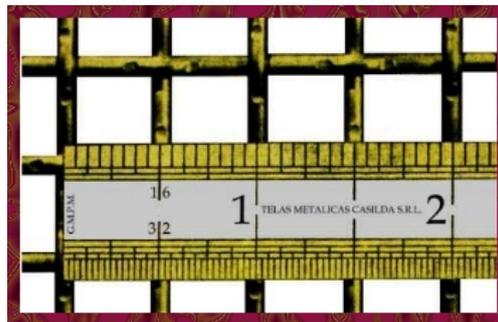


Figura A-1: Medición de la Malla – Ej. Tela de 2 mallas por pulgada.

Abertura o Luz de la Malla: La luz de malla o abertura es la luz libre expresada en milímetros entre dos alambres contiguos. Su tamaño no es afectado por el mayor o menor diámetro del alambre.

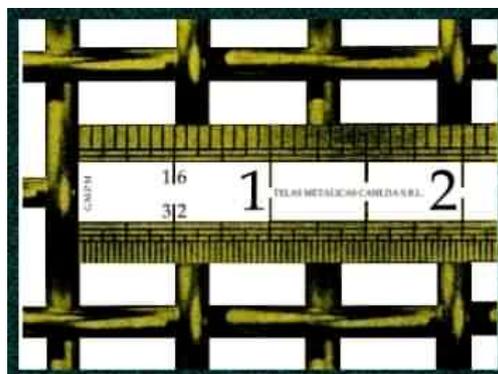


Figura A-2: Medición de la Abertura – Ej. Abertura de 12.7 mm = 1/2 pulgada

La Abertura puede ser calculada como:

$$Abertura = \frac{25.4}{n_{malla}} - d_{alambre} \quad [mm]$$

siendo n_{malla} el número de malla por pulgada definida más arriba y $d_{alambre}$ es el diámetro del alambre que conforma la tela metálica.

Por ejemplo, la empresa¹⁰ *Telas Metálicas Casilda S.R.L.*, de Loma Hermosa, Bs. As., fabrica entre otras las mallas especificadas en la siguiente tabla:

Tabla A-1: Especificaciones Mallas Finas Standard

MALLA	DIA. ALAMBRE	ABERTURA	AREA ABIERTA
/pulg.	mm	mm	%
20	0.30	0.97	58.34
25	0.25	0.77	56.84
30	0.30	0.55	41.69
40	0.20	0.44	46.93
50	0.20	0.31	36.76
60	0.18	0.24	33.04
100	0.10	0.15	36.76

Área Abierta: Es la fracción de superficie de la tela no cubierta por los alambres. Se calcula de la siguiente manera:

$$A_{Abierta} = \frac{S_{Abierta}}{S_{Total}} = \left(\frac{Abertura \cdot n_{malla}}{25.4} \right)^2 [-]$$

Desde el punto de vista comercial la malla queda definida especificando el número de malla, el diámetro de alambre y el tipo de material deseado (Acero Inoxidable, Bronce, Cobre, etc.).

A-2. Longitud de los alambres

Debido a la particular geometría del entramado de una malla de alambres, la longitud de los alambres, c , es mayor que la longitud de la celda, l , tal como se puede observar en la Figura A-3.

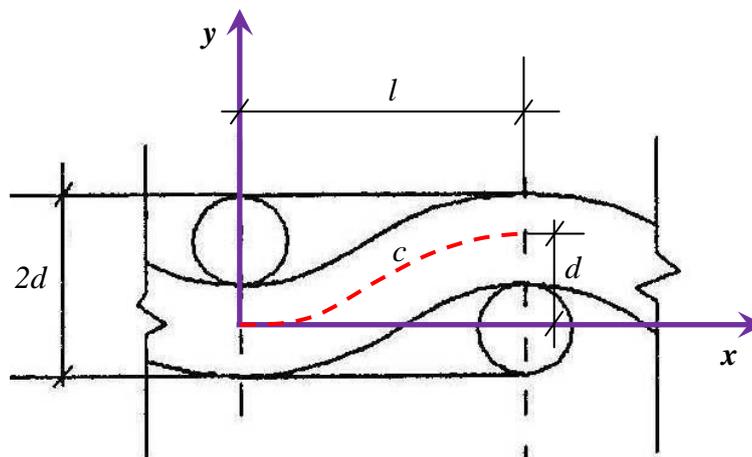


Figura A-3: Longitud de los alambres

Según se observa en la Figura A-3, la ecuación que describe la línea c puede escribirse como:

$$y = -2 \cdot d_{alambre} \cdot \left(\frac{x}{l}\right)^3 + 3 \cdot d_{alambre} \cdot \left(\frac{x}{l}\right)^2$$

siendo l la longitud de la celda, dado por: $l = 25.4/n_{malla} [mm]$.

La longitud de la curva c puede ser obtenido resolviendo la integral

$$Long(c) = \int_0^l \|\alpha'(x)\| dx \quad \text{con } \alpha(x) = (x, y).$$

Conocido el factor f , el cual se define como: $f = \frac{Long(c)}{l} [-]$

es posible calcular la longitud de los alambres para cualquier tipo de malla (ver Tabla A-2).

Una ecuación aproximada que permite obtener f es propuesta por Ref. [9]:

$$f \approx 1 + 9.6 \cdot \left(\frac{d_{alambre}}{4} \cdot \frac{n_{malla}}{25.4}\right)^2 - 49.2 \cdot \left(\frac{d_{alambre}}{4} \cdot \frac{n_{malla}}{25.4}\right)^4$$

A-3. Cálculo del espesor de la matriz

Por otro lado, la longitud de la matriz depende del diámetro de los alambres (Fig. A-4) y del tipo de apilamiento que se haga de las telas, es decir que las mismas estén alineadas o no, tal como se muestra en la Figura A-5.

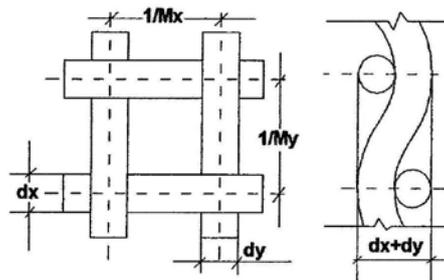


Figura A-4: Espesor de la tela de alambres

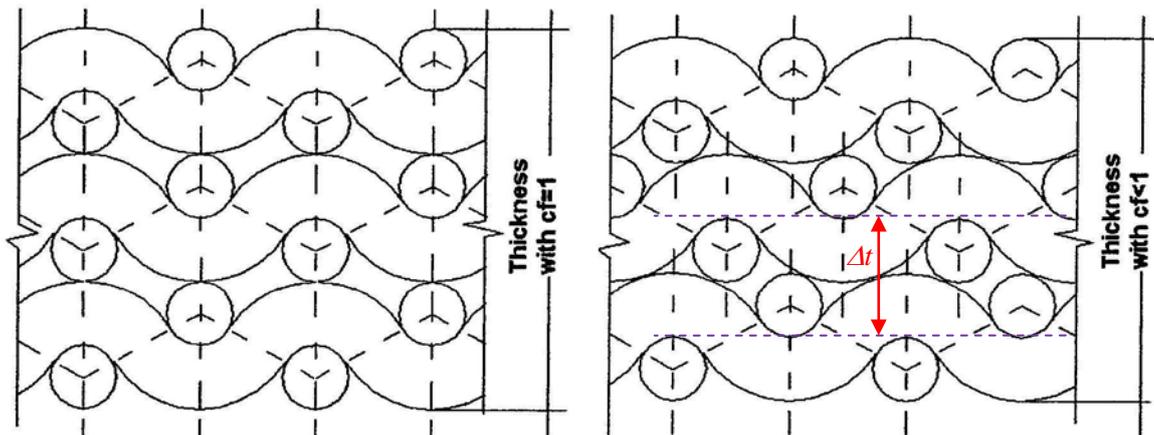


Figura A-5: Geometrías de telas apiladas para diferentes factores de compresión (cf)



Para telas con igual diámetro para los alambres que conforman la urdimbre y la trama², $d_x = d_y = d_{alambre}$, el espesor de la matriz (Fig. A-3) se calcula como:

$$t = 2 \cdot d_{alambre} \cdot n_{telas} \cdot cf$$

A-4. Factor de Compresión

El factor de compresión, cf , si la malla no sufre aplastamiento posee un valor que se encuentra entre un máximo ($cf=1$), cuando las mallas está desfasadas media longitud de celda, a un mínimo cuando se encuentran desfasada un cuarto de longitud de celda (ver Figura A-5).

El factor de compresión puede ser obtenido haciendo:

$$cf = \frac{2 \cdot d_{alambre} + (n_{telas} - 1) \cdot \Delta t}{2 \cdot d_{alambre} \cdot n_{telas}}$$

siendo Δt la distancia definida en la Figura A-5

Se observa que para n_{telas} grande el cf mínimo converge a: $cf|_{n_{telas} \rightarrow \infty} = \frac{\Delta t}{2 \cdot d_{alambre}}$

En la siguiente tabla se presentan los valores Δt para las mallas definidas en Tabla A-1. Los mismos fueron obtenidos gráficamente mediante el uso de CATIA[®].

Tabla A-2: Factores de Compresión

MALLA	DIA. ALAMBRE	Δt	cf_{min}
/pulg.	mm	mm	%
20	0.30	0.336	55.97
25	0.25	0.284	56.73
30	0.30	0.378	63.03
40	0.20	0.243	60.68
50	0.20	0.262	65.45
60	0.18	0.243	67.38
100	0.10	0.130	65.09

A-5. Cálculo de la porosidad

Por otro lado es necesario definir los parámetros con que se especifica la matriz en el programa StirSimple.for del ANEXO B. Por ello se define:

² Los alambres que corren paralelos (longitudinales) a lo largo de la tela se denominan *URDIMBRES* y los que corren a lo ancho de la misma (transversales) se denominan *TRAMAS*.

Porosidad: es la relación entre el volumen del espacio vacío (u ocupado por un fluido) V_V , respecto del volumen total (incluyendo el espacio ocupado por el material y el vacío) V_T . La porosidad es indicada normalmente con las letras ϕ , n o ε .

$$\phi = \frac{V_V}{V_T}$$

La porosidad es una fracción entre 0 y 1, por lo general van desde menos de 0.01 para el granito sólido a más de 0.5 para la turba y arcilla. También puede ser expresada en porcentaje multiplicando la fracción por 100.

Para una tela de alambre el valor de la porosidad se puede calcular como:

$$\phi = \frac{V_V}{V_T} = \frac{t \cdot 25.4^2 - 2 \cdot n_{\text{malla}} \cdot \left(\pi \cdot \frac{d_{\text{alambre}}^2}{4} \cdot 25.4 \cdot f \right) \cdot n_{\text{telas}}}{t \cdot 25.4^2} = 1 - \frac{\pi}{4} \cdot d_{\text{alambre}} \cdot \frac{n_{\text{malla}}}{25.4} \cdot \frac{f}{cf} [-]$$

Otro tipo de apilamiento puede producir un gran aumento de la caída de presión (ver Fig. A-5).

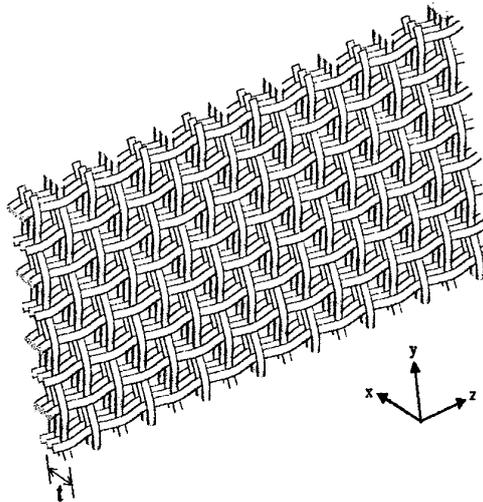


Figura A-6: Geometría de la Matriz formada por telas metálicas alineadas

A-6. Cálculo de la superficie mojada

La relación entre la superficie mojada de una tela, respecto de la superficie de la misma vista en planta resulta:

$$\frac{S_w}{S} = \frac{2 \cdot n_{\text{malla}} \cdot \left(\pi \cdot d_{\text{alambre}} \cdot 25.4 \cdot f \right) \cdot n_{\text{telas}}}{25.4^2} = 2 \cdot \pi \cdot d_{\text{alambre}} \cdot \frac{n_{\text{malla}}}{25.4} \cdot f \cdot n_{\text{telas}} [-]$$

Comparando esta expresión con la obtenida para la porosidad se observa que:

$$\phi = 1 - \frac{S_w/S}{8 \cdot n_{\text{telas}} \cdot cf}$$

La superficie mojada por unidad de volumen es conocida como *Superficie Mojada Específica*, β , y se calcula como:



$$\beta = \frac{S_w}{V_T} = \frac{S_w/S}{t} = 4 \frac{(1-\phi)}{d_{alambre}} = \pi \cdot \frac{n_{malla}}{25.4} \cdot \frac{f}{cf} \quad [1/mm]$$

A-7. Cálculo del Diámetro Hidráulico

El diámetro hidráulico, D_h , es un término comúnmente utilizado en hidráulica cuando se manejan fluidos en tubos no circulares. Utilizando éste término se puede estudiar el comportamiento del flujo de la misma forma como si fuera una tubería de sección circular. Por definición:

$$D_h = \frac{4 \cdot V_V}{S_w} = \frac{4 \cdot \phi}{\beta} = \frac{\phi}{(1-\phi)} \cdot d_{alambre} = \frac{4}{\pi} \cdot \frac{25.4}{n_{malla}} \cdot \frac{cf}{f} \cdot d_{alambre} \quad [mm]$$

Para presentarlo en forma adimensional, para el caso de una malla de alambre, conviene escribir:

$$\frac{D_h}{d_{alambre}} = \frac{\phi}{(1-\phi)} = \frac{4}{\pi} \cdot \frac{25.4}{n_{malla} \cdot d_{alambre}} \cdot \frac{cf}{f} - 1 \quad [-]$$

Tabla A-3: Características de las Mallas Finas Standard ($n_{telas}=1$)

MALLA	f	S_w/S	ϕ	β	$D_h/d_{alambre}$
/pulg.	-	-	%	1/mm	-
20	1.0327	1.5328	80.840	2.5546	4.2193
25	1.0354	1.6008	79.989	3.2017	3.9974
30	1.0717	2.3859	70.176	3.9766	2.3530
40	1.0572	2.0922	73.848	5.2304	2.8238
50	1.0876	2.6904	66.371	6.7259	1.9736
60	1.1012	2.9421	63.224	8.1725	1.7192
100	1.0876	2.6904	66.371	13.4518	1.9736

Por ej., en 1 m² de tela para 1 malla de 60 (Tabla A-1), la superficie mojada es $S_w = 2.9421$ m².

Conocida la superficie mojada necesaria y la superficie frontal de la matriz, con la ecuación anterior se puede estimar el número de telas, n_{telas} , que la conforman.



ANEXO B: MANUAL DE USUARIO – PROGRAMA STIRSIMPLE

B-1 INTRODUCCION

El programa StirSimple es un programa en Fortran que permite calcular las performances de un motor Stirling asumiendo el modelo llamado Simple desarrollado por el Dr. Israel Urieli (Ref. [2]). El modelo matemático fue documentado en la primera parte de este informe. En este Anexo se presenta el listado fuente del programa junto con un archivo de entrada de ejemplo y su correspondiente archivo de salida.

B-2 ARCHIVO DE ENTRADA

El archivo de entrada StirSimple.dat consta de diez Namelist, con la siguiente información:

1. **&Motor**, datos generales de los pistones

- 1.1. **vswc**, Volumen de barrido de compresión (zona fría) [m³]
- 1.2. **vswe**, Volumen de barrido de expansión (zona caliente) [m³]
- 1.3. **vcic0**, Volumen libre de compresión (volumen libre de la zona fría) [m³]
- 1.4. **vcle0**, Volumen libre de expansión (volumen libre de la zona caliente) [m³]
- 1.5. **phase**, Desfasaje de volumen [°]

Nota: los volúmenes libres no incluyen los conductos de unión.

2. **&Inter**, Define los tipos de intercambiadores de calor y matriz del regenerador.

- 2.1. **t_enfriador**, Tipo de enfriador: t)ubos a)nillo r)anurado m)alla de alambres
- 2.2. **t_calentador**, Tipo de calentador: t)ubos a)nillo r)anurado m)alla de alambres
- 2.3. **t_regenerador**, Tipo de regenerador: t)ubular a)nular
- 2.4. **t_matriz**, Tipo de matriz: m)alla de alambres l)ámina a)lambres apilados s)in matriz

3. **&Geom_Enf**, Datos geométricos del enfriador. Los datos de entrada dependen del tipo seleccionado en §2.1. A modo de ejemplo se presenta el tipo m).

- 3.1. **din**, Diámetro interno del tubo [m]
- 3.2. **len**, Longitud del intercambiador de calor [m]
- 3.3. **dwire**, Diámetro de los alambres de la malla de alambre [m]
- 3.4. **porosity**, Porosidad de la tela de alambres (ver § A-5)



4. **&Geom_Cal**, Datos geométricos del calentador. Los datos de entrada dependen del tipo seleccionado en §2.2. A modo de ejemplo se presenta el tipo r).
 - 4.1. **w**, Ancho de la ranura [m]
 - 4.2. **h**, Alto de la ranura [m]
 - 4.3. **len**, Longitud del intercambiador de calor [m]
 - 4.4. **num**, Número de ranuras del paquete

5. **&Geom_Reg**, Datos geométricos del regenerador. Los datos de entrada dependen del tipo seleccionado en §2.3. A modo de ejemplo se presenta el tipo t).
 - 5.1. **dout**, Diámetro externo del cárter del tubo [m]
 - 5.2. **domat**, Diámetro interno del cárter del tubo [m]
 - 5.3. **lr**, Longitud del regenerador [m]
 - 5.4. **num**, Número de tubos

6. **&Geom_Malla**, Datos geométricos de la malla del regenerador. Los datos de entrada dependen del tipo seleccionado en §2.4. A modo de ejemplo se presenta el tipo m).
 - 6.1. **dwire**, Diámetro de los alambres de la malla de alambre [m]
 - 6.2. **porosity**, Porosidad de la tela de alambres (ver § A-5)

7. **&Geom_Conk**, Datos geométricos de la conexión enfriador-regenerador.
 - 7.1. **dck**, Diámetro hidráulico del tubo del enfriador [m]
 - 7.2. **lck**, Longitud efectiva del tubo del enfriador [m]

8. **&Geom_Conh**, Datos geométricos de la conexión calentador-regenerador.
 - 8.1. **dch**, Diámetro hidráulico del tubo del calentador [m]
 - 8.2. **lch**, Longitud efectiva del tubo del calentador [m]

9. **&Tipo_Gas**, Tipo de gas de trabajo.
 - 9.1. **t_gas**, Tipo de gas de trabajo: ai)re, he)lio, hi)drógeno

10. **&Operacion**, Datos geométricos de la conexión calentador-regenerador.
 - 10.1. **pmean**, Presión promedio [Pa]
 - 10.2. **tk**, Temperatura externa en el enfriador [K]
 - 10.3. **th**, Temperatura externa en el calentador [K]
 - 10.4. **freq**, Velocidad de rotación de la manivela [Hz]



Archivo de entrada ejemplo (StirSimple.dat):

---- INTIHUASI - Concepto ----

```
&Motor
  vswc = 98.17D-6
  vswe = 98.17D-6
  vclc0 = 20.11008D-6
  vcle0 = 41.79460D-6
  phase = 90.0D+0
/
&Inter
  t_enfriador = 'm'
  t_calentador = 'm'
  t_regenerador = 't'
  t_matriz = 'm'
/
&Geom_Enf
  din = 60.00D-3
  len = 6.131D-3
  dwire = 0.180D-3
  porosity = 0.63226D0
/
&Geom_Conk
  dck = 12.D-3
  lck = 125.132D-3
/
&Geom_Reg
  dout = 92.00D-3
  domat = 82.00D-3
  lr = 19.59D-3
  num = 1
/
&Geom_Malla
  porosity = 0.63226D0
  dwire = 0.18D-3
/
&Geom_Conh
  dch = 17.0D-3
  lch = 57.0D-3
/
&Geom_Cal
  din = 100.0D-3
  len = 4.756D-3
  dwire = 0.180D-3
  porosity = 0.63226D0
/
&Tipo_Gas
  t_gas = 'ai'
/
&Operacion
  pmean = 100000.D0
  tk = 300.D0
  th = 923.D0
  freq = 50.D0
/
```



B-3 DATOS POR PANTALLA

Cuando se corre el programa aparece por pantalla la siguiente información:

1. Todos los namelist con los datos de entrada.
2. En el caso de seleccionarse matrices compuestas por mallas de alambre, se muestra la cantidad de mallas para cada intercambiador.
3. Cuadro conteniendo las performances del motor de acuerdo al *Análisis Isotérmico de Schmidt* (Ref. [2]). Ejemplo:

```
----- Analisis de Schmidt -----  
Trabajo [joule]      3.0567   Potencia [watts]   152.8339  
Qexpans [joule]     4.5286   Qcompres [joule]  -1.4719  
Eficiencia Indicada [%]      67.4973  
-----
```

Donde

- 3.1. **Trabajo [joule]**, Trabajo realizado por ciclo ($Q_{\text{expansión}}+Q_{\text{compresión}}$)
 - 3.2. **Potencia [watts]**, Potencia entregada (Trabajo·freq)
 - 3.3. **Qexpans [joule]**, Calor transferido en la zona de expansión por ciclo (zona caliente)
 - 3.4. **Qcompres [joule]**, Calor transferido en la zona de compresión por ciclo (zona fría)
 - 3.5. **Eficiencia Indicada [%]**, Eficiencia del ciclo (Trabajo/ $Q_{\text{expansión}}$)
4. Ingreso del tipo de simulación deseada a)diabática o s)imple.
 5. Si la simulación es de tipo adiabática (Ref. [1]) se muestra un cuadro conteniendo información sobre los resultados obtenidos con este tipo de modelo. Ejemplo:

```
----- Analisis Adiabatico Ideal -----  
Calor transferido al enfriador [W]: -102.3908  
Calor transferido al regenerador [W]: 0.0023  
Calor transferido al calentador [W]: 266.9577  
Potencia total de salida [W]: 164.7670  
Eficiencia térmica del motor [%]: 61.7202  
-----
```

Donde

- 5.1. **Calor transferido al enfriador [W]**, Calor entrante (+) o saliente (-) de la zona de enfriamiento por segundo.
 - 5.2. **Calor transferido al regenerador [W]**, Ídem anterior para la zona del regenerador
 - 5.3. **Calor transferido al calentador [W]**, Ídem anterior para la zona del calentador
 - 5.4. **Potencia total de salida [W]**, Es la potencia total entregada (+) o absorbida (-) por ambos pistones.
 - 5.5. **Eficiencia térmica del motor [%]**, Eficiencia del ciclo (Potencia total de salida/Calor transferido al calentador)
6. Las siguientes tablas se muestran tantas veces como iteraciones son necesarias para estabilizar las temperaturas (ver §2.2). En la primera de ellas se presentan las performances del regenerador. Ejemplo:



----- Analisis Simple del Regenerador -----

Numero de Reynolds promedio	:	15.8812
Numero de Reynolds máximo	:	31.2404
Numero de Staton (a Re promedio)	:	0.2144
Numero de unidades de transferencia	:	27.3003
Eficiencia del regenerador [%]	:	96.4665
Perdida de Entalpia neta del reg.[W]	:	43.7398

Donde

- 6.1. **Numero de Reynolds promedio**, Número de Reynolds promedio en la zona del regenerador usado para el cálculo del Número de Staton, NTU, eficiencia y perdida de entalpia del regenerador.
 - 6.2. **Numero de Reynolds máximo**, permite conocer la amplitud de variación del N_{Re} .
 - 6.3. **Numero de Staton**, basado en el número de Reynolds promedio del regenerador y dependiente del tipo de matriz seleccionada, el número de Staton promedio es usado para el cálculo del número de unidades de transferencia (NTU).
 - 6.4. **Numero de unidades de transferencia**, ver §2.1.2
 - 6.5. **Eficiencia del regenerador**, valor obtenido en base al NTU.
 - 6.6. **Perdida de Entalpia neta del reg.**, calor pedido por segundo en la zona del regenerador.
7. Tabla conteniendo los resultados de performances del Calentador de acuerdo al modelo Simple. Ejemplo:

----- Analisis Simple del Calentador -----

Numero de Reynolds promedio	:	6.5393
Numero de Reynolds máximo	:	13.3135
Coef. de transf. de calor [W/K/m ²]	:	28.5271
Temp. de pared del calentador [K]	:	923.0000
Temp. del gas en el calentador [W]	:	888.1929

Donde

- 7.1. **Numero de Reynolds promedio**, Usado para el cálculo del factor de fricción de Reynolds, basado en la relación de Blasius, y del coeficiente de transferencia de calor.
 - 7.2. **Numero de Reynolds máximo**, permite conocer la amplitud de variación del N_{Re} .
 - 7.3. **Coef. de transf. de calor**, coeficiente de transferencia de calor, obtenido a partir de la analogía simple de Reynolds.
 - 7.4. **Temp. de pared del calentador**, temperatura de la pared exterior del calentador, ingresada como dato de entrada.
 - 7.5. **Temp. del gas en el calentador**, temperatura del gas dentro del calentador. Obtenida del cálculo iterativo propuesto en §2.2.
8. Tabla conteniendo los resultados de performances del Enfriador de acuerdo al modelo Simple. Ejemplo:

----- Analisis Simple del Enfriador -----

Numero de Reynolds promedio	:	57.0148
Numero de Reynolds máximo	:	109.2497



Coef. de transf. de calor [W/K/m²] : 71.5872
Temp. de pared del enfriador [K] : 300.0000
Temp. del gas en el enfriador [W] : 314.9087

Donde

- 8.1. **Numero de Reynolds promedio**, Usado para el cálculo del factor de fricción de Reynolds, basado en la relación de Blasius, y del coeficiente de transferencia de calor.
 - 8.2. **Numero de Reynolds máximo**, permite conocer la amplitud de variación del N_{Re} .
 - 8.3. **Coef. de transf. de calor**, coeficiente de transferencia de calor, obtenido a partir de la analogía simple de Reynolds.
 - 8.4. **Temp. de pared del enfriador**, temperatura de la pared exterior del enfriador, ingresada como dato de entrada.
 - 8.5. **Temp. del gas en el enfriador**, temperatura del gas dentro del enfriador. Obtenida del cálculo iterativo propuesto en §2.2.
9. Resultados del análisis con el modelo adiabático como parte del modelo Simple. Ejemplo:

----- Analisis Adiabatico Ideal Simple -----
Calor transferido al enfriador [W]: -108.6853
Calor transferido al regenerador [W]: 0.0048
Calor transferido al calentador [W]: 260.8435
Potencia total de salida [W]: 152.3840
Eficiencia térmica del motor [%]: 58.4197

Donde

- 9.1. **Calor transferido al enfriador [W]**, Calor entrante (+) o saliente (-) de la zona de enfriamiento por segundo.
 - 9.2. **Calor transferido al regenerador [W]**, Ídem anterior para la zona del regenerador
 - 9.3. **Calor transferido al calentador [W]**, Ídem anterior para la zona del calentador
 - 9.4. **Potencia total de salida [W]**, Es la potencia total entregada (+) o absorbida (-) por ambos pistones.
 - 9.5. **Eficiencia térmica del motor [%]**, Eficiencia del ciclo (Potencia total de salida/Calor transferido al calentador)
10. Tabla conteniendo los datos más relevantes del análisis del Regenerador. Ejemplo:

----- Analisis Resultados del Regenerador -----
Entalpia neta perdida [W]: 43.7398
Fuga de calor en la pared [W]: 13.0381

Donde

- 10.1. **Perdida de Entalpia neta del reg.**, entalpía perdida por segundo en la zona del regenerador. Dato idéntico al presentado en la tabla Análisis Simple del Regenerador, punto 6.6
- 10.2. **Fuga de calor en la pared**, calor perdido por el regenerador. Este valor es función de la conductividad térmica de la pared del carter del regenerador el cual a su vez depende del tipo de material del carter (pared y aislante térmico) (ver SUBROUTINE Ther_Cond).



11. Tabla de resultados del motor Stirling obtenidos con el modelo Simple. Ejemplo:

```
----- Analisis Simple por Perdidas -----  
Trab.perdido por perdida de Pres.[W]:      20.2191  
Calor real transf. al calentador [W]:      317.6214  
Potencia real                               [W]:      132.1649  
Eficiencia termica real del motor[%]:      41.6108  
-----
```

Donde

- 11.1. **Trab.perdido por perdida de Pres.**, trabajo perdido por segundo debido a la pérdida de presión. Ver la teoría en §2.3 y la implementación en la SUBROUTINE SimWork(dwork).
- 11.2. **Calor real transf. al calentador**, el calor real transferido al calentador se obtiene de sumar el calor ingresante obtenido del modelo adiabático más la entalpía y el calor perdido en el regenerador.
- 11.3. **Potencia real**, es la potencia real entregada por el motor. La misma es obtenida restando a la potencia calculada por el modelo adiabático la potencia perdida por pérdida de presión.
- 11.4. **Eficiencia termica real del motor**, es la eficiencia real del motor y es obtenida dividiendo la potencia real por el calor real transferido al calentador.

B-4 ARCHIVOS DE SALIDA

Se generan 3 archivos de salida con los datos presentados en columnas para ser leídos fácilmente desde Excel®. Por cuestiones de espacio las tablas resultado de ejemplo se muestran en forma comprimida y luego de ser levantada a una planilla Excel. Seguidamente se describen cada uno de ellos.

1. **SchmidtPV.grf**, gráficos p-v y p-theta del análisis de Schmidt

- 1.1. Theta [gr], Angulo de manivela (de 0° a 360° con paso de 5°)
- 1.2. Vc [cm³], Volumen de la zona de compresión (zona fría)
- 1.3. Ve [cm³], Volumen de la zona de expansión (zona caliente)
- 1.4. Vtot [cm³], Volumen total (Vc+Vk+Vr+Vh+Ve)
- 1.5. P [Bar], Presión en el motor
- 1.6. Pmean [Bar], Presión media

Archivo de salida ejemplo (SchmidtPV.grf):



Theta [gr]	Vc [cm^3]	Ve [cm^3]	Vtot [cm^3]	P [Bar]	Pmean [Bar]
0	132.43218	103.81746	336.23755	0.74091	1
5	132.24539	99.53942	331.77273	0.74626	1
10	131.68646	95.29394	326.96832	0.75293	1
15	130.75964	91.11333	321.86089	0.76096	1
20	129.47199	87.02941	316.4893	0.77036	1
170	35.00789	95.29394	230.28974	1.35946	1
175	34.44896	99.53942	233.97629	1.35022	1
180	34.26218	103.81746	238.06755	1.3369	1
185	34.44896	108.0955	242.53237	1.31985	1
190	35.00789	112.34098	247.33678	1.29947	1
340	129.47199	120.60552	350.06542	0.73261	1
345	130.75964	116.5216	347.26915	0.73274	1
350	131.68646	112.34098	344.01536	0.73416	1
355	132.24539	108.0955	340.32881	0.73688	1
360	132.43218	103.81746	336.23755	0.74091	1

2. **SchmidtMass.grf**, gráfico de Mapa de Trayectoria de Partículas (Ref. [11]).

- 2.1. Theta [gr], Angulo de manivela (de 0° a 360° con paso de 5°)
- 2.2. L1, Límite del volumen de barrido de expansión reducido
- 2.3. L2, Límite del volumen libre de expansión reducido
- 2.4. L3, Límite del volumen libre del calentador reducido
- 2.5. L4, Límite del volumen libre del regenerador reducido
- 2.6. L5, Límite del volumen libre del enfriador reducido
- 2.7. L6, Límite del volumen de barrido de compresión reducido
- 2.8. plot_v, 31 segmentos de volúmenes reducidos (step=30)

Archivo de salida ejemplo (SchmidtMass.grf):

Theta [gr]	L1	L2	L3	L4	L5	L6	v0	v1	v29	v30
0	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0	0.06352	1.84215	1.90567
5	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.00062	0.06557	1.8843	1.94925
10	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.00247	0.0688	1.92616	1.9925
15	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.00554	0.07319	1.96743	2.03508
20	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.0098	0.0787	2.00779	2.07668
170	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.32256	0.37822	1.93683	1.9925
175	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.32441	0.37857	1.89509	1.94925
180	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.32503	0.37772	1.85298	1.90567
185	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.32441	0.37566	1.81084	1.86209
190	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.32256	0.37243	1.76897	1.81885
340	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.0098	0.0673	1.67717	1.73466
345	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.00554	0.06456	1.71724	1.77626
350	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.00247	0.06301	1.7583	1.81885
355	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0.00062	0.06267	1.80004	1.86209
360	0.32503	0.50624	0.58443	0.94502	1.05666	1.40567	0	0.06352	1.84215	1.90567

3. **PlotLoss.grf**, gráficos de pérdida de presión y calor.

- 3.1. Theta [gr], Angulo de manivela (de 0° a 360° con paso de 10°)



- 3.2. dPck [Pa], Pérdida de presión en el tubo de conexión enfriador-regenerador
- 3.3. dPkol [Pa], Pérdida de presión en el enfriador
- 3.4. dPreg [Pa], Pérdida de presión en el regenerador
- 3.5. dPhot [Pa], Pérdida de presión en el calentador
- 3.6. dPch [Pa], Pérdida de presión en el tubo de conexión enfriador-regenerador
- 3.7. dPtot [Pa], Pérdida de presión total (dPck+dPkol+dPreg+dPhot+dPch)
- 3.8. Padiab [Bar], Presión en el motor según el Modelo Adiabático (Ref. [1])
- 3.9. Psimple [Bar], Presión en el motor según el Modelo Simple (Padiab+dPtot)

Archivo de salida ejemplo (PlotLoss.grf):

Theta [gr]	dPck [Pa]	dPkol [Pa]	dPreg [Pa]	dPhot [Pa]	dPch [Pa]	dPtot [Pa]	Padiab [Bar]	Psimple [Bar]
0	25.21563	132.05026	1062.09587	525.14193	71.09463	1815.59832	1.39192	1.41007
10	-15.00713	-36.30656	486.88131	395.18944	47.5999	878.35696	1.34417	1.35295
20	-120.73653	-237.6447	-44.5275	249.76825	25.21978	-127.92069	1.28375	1.28247
30	-273.65167	-466.74294	-564.54798	107.14463	8.80968	-1188.98827	1.21519	1.2033
40	-431.85506	-687.75852	-1094.85768	-24.40084	0.5468	-2238.3253	1.14311	1.12073
160	-260.82134	-626.68577	-2370.59163	-795.3138	-83.17095	-4136.58349	0.69386	0.6525
170	-179.33955	-491.4909	-2074.49371	-752.36278	-77.94675	-3575.63368	0.69884	0.66309
180	-106.10254	-353.74127	-1739.17564	-693.56814	-70.60333	-2963.19091	0.70993	0.6803
190	-46.26279	-217.93954	-1371.50353	-619.81972	-61.38318	-2316.90876	0.72725	0.70408
200	-7.30693	-90.04997	-980.11359	-532.32289	-50.63423	-1660.42761	0.75099	0.73439
320	693.3717	1044.24493	2817.6971	724.4415	101.13945	5380.89467	1.40844	1.46225
330	514.05139	840.07244	2549.32024	732.83167	107.66163	4743.93738	1.43305	1.48049
340	316.91725	597.62501	2137.06783	700.10325	104.12309	3855.83643	1.43844	1.477
350	141.3561	350.47081	1623.41918	628.59695	91.04759	2834.89063	1.42425	1.4526
360	25.21563	132.05026	1062.09587	525.14193	71.09463	1815.59832	1.39192	1.41007

B-5 LISTADO FUENTE

Seguidamente se lista el programa fuente y sus subrutinas. Las variables son definidas mediante un include.

PROGRAMA PRINCIPAL: StirSimple

```

*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema       : Calculo de las performances de un motor Stirling                               **
**            calculando las pérdidas de presión                                           **
** Descripcion: Modelo adiabático corregido por pérdidas de presión                       **
**            y transferencia de calor                                                       **
**            Modelo basado en el desarrollo del Dr. Israel Urieli                         **
*****
** Estructura del programa:                                                                 **
**                                                                                          **
** 1   StirSimple                                                                           **
** 1.1 define                                                                               **
** 1.1.1 engine                                                                             **
** 1.1.1.1   sindrive                                                                     **
** 1.1.2   heatex                                                                           **

```



```
** 1.1.2.1 cooler **
** 1.1.2.1.1 pipes **
** 1.1.2.1.2 annulus **
** 1.1.2.1.3 slots **
** 1.1.2.1.4 tela **
** 1.1.2.2 heater **
** 1.1.2.2.1 pipes **
** 1.1.2.2.2 annulus **
** 1.1.2.2.3 slots **
** 1.1.2.2.4 tela **
** 1.1.2.3 regen **
** 1.1.2.3.1 tubular **
** 1.1.2.3.2 annular **
** 1.1.2.3.3 matrix **
** 1.1.2.3.3.1 mesh **
** 1.1.2.3.3.2 foil **
** 1.1.2.3.3.3 alambre **
** 1.1.2.3.3.4 nomatrix **
** 1.1.2.3.4 ther_cond **
** 1.1.3 gas **
** 1.1.3.1 air **
** 1.1.3.2 helium **
** 1.1.3.3 hydrogen **
** 1.1.4 operat **
** 1.2 Schmidt **
** 1.2.1 plotpv **
** 1.2.2 plotmass **
** 1.3 adiabatic **
** 1.3.1 adiab **
** 1.3.1.1 rk4 **
** 1.3.1.1.1 dadiab **
** 1.3.1.1.1.1 volume **
** 1.3.1.1.1.1.1 sinevol **
** 1.3.1.2 filmatrix **
** 1.3.2 plotadiab **
** 1.4 simple **
** 1.4.1 adiab **
** 1.4.1.1 rk4 **
** 1.4.1.1.1 dadiab **
** 1.4.1.1.1.1 volume **
** 1.4.1.1.1.1.1 sinevol **
** 1.4.2 SimReg **
** 1.4.2.1 reynum **
** 1.4.2.2 frmatrix **
** 1.4.2.3 frfoil **
** 1.4.2.4 fralambre **
** 1.4.3 SimHot **
** 1.4.3.1 reynum **
** 1.4.3.2 frpipe **
** 1.4.4 SimKol **
** 1.4.4.1 reynum **
** 1.4.4.2 frpipe **
** 1.4.5 SimWork **
** 1.4.5.1 reynum **
** 1.4.5.2 frmatrix **
** 1.4.5.3 frfoil **
** 1.4.5.4 frpipe **
** 1.4.5.5 fralambre **
** 1.4.5.6 plotloss **
** 1.4.6 plotadiab **
*****
** Programa Principal : StirSimple.for **
** Autor : Gustavo Scarpin **
** Fecha : 31 de Mayo de 2012 **
** Ultima modificación: 06 de Octubre de 2012 **
** Modificación: Incorporación de las perdidas de calor y presión en **
** los volúmenes vclc y vcle **
*****
```

PROGRAM StirSimple

C Definir las variables locales
CHARACTER*1 tipo_anal

OPEN(UNIT=3,FILE='StirSimple.out',STATUS='UNKNOWN')



```
CALL Define
CALL Schmidt

C INGRESA EL TIPO DE ANALISIS QUE SE QUIERE REALIZAR
C a)diabatiko s)imple
WRITE(*,1000)
READ(*,'(A1)') tipo_anal

IF (tipo_anal.EQ.'a') THEN
CALL Adiabatic
ELSE IF (tipo_anal.EQ.'s') THEN
CALL Simple
ELSE
STOP 'FIN DEL PROGRAMA StirSimple'
ENDIF

1000 FORMAT(2X,'Tipo de simulacion a)diabatika o s)simple ')

END PROGRAM StirSimple
```

SUBROUTINAS

```
*****
**          Modelo Simple Análisis de un motor Stirling          **
*****
** Tema      : Simulación de un motor Stirling mediante un modelo **
**          : adiabático ideal **
** Descripcion: **
*****
** Subroutine : Adiab.for **
** Programa Principal : StirSimple.for **
** Autor      : Gustavo Scarpin **
** Fecha     : 3 de Junio de 2012 **
** Ultima modificación: 3 de Junio de 2012 **
*****
```

```
SUBROUTINE Adiab
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definición de parámetros y variables locales de simulación
REAL*8, PARAMETER :: epsilon = 0.01D0 !Error de temperatura admisible [K]
INTEGER, PARAMETER :: max_iteration = 20 !Número máximo de iteraciones para convergencia
INTEGER, PARAMETER :: ninc = 360 !Número de incrementos de integración (cada grado)
INTEGER, PARAMETER :: step = 10 !Cantidad de valores a salvar en las matrices var y dvar
REAL*8, PARAMETER :: dtheta = 2.0D*pi/ninc !Incrementos de integración [rad]
REAL*8 terror,theta,tc0,te0
INTEGER iter,i,j

REAL*8 y(ROWV)
REAL*8 dy(ROWD)

C Condiciones iniciales
y(THE) = th
y(TCK) = tk
y(TE) = th
y(TC) = tk
iter = 0
terror = 10.0D*epsilon !Error inicial a la entrada del ciclo de convergencia

C Ciclo de convergencia
DO WHILE (terror.GE.epsilon.AND.iter.LT.max_iteration)

C Condiciones iniciales del ciclo
tc0 = y(TC)
te0 = y(TE)
theta = 0.0D
y(QK) = 0.0D
y(QR) = 0.0D
y(QH) = 0.0D
y(WC) = 0.0D
y(WE) = 0.0D
y(W) = 0.0D

DO i=1,ninc
```



```
CALL rk4(7,theta,dtheta,y,dy,ROWV,ROWD)
END DO

terror = DABS(tc0 - y(TC)) + DABS(te0 - y(TE))
iter = iter + 1

END DO

C Iniciación de las matrices var y dvar
var = 0.D0
dvar = 0.D0

C Ciclo final para cargar las matrices var y dvar
theta = 0.D0
y(QK) = 0.D0
y(QR) = 0.D0
y(QH) = 0.D0
y(WC) = 0.D0
y(WE) = 0.D0
y(W) = 0.D0

CALL Filmatrix(1,y,dy)

DO i=2,COL
DO j=1,step
CALL rk4(7,theta,dtheta,y,dy,ROWV,ROWD)
END DO
CALL Filmatrix(i,y,dy)
END DO

END SUBROUTINE Adiab

*****
** Subroutine: Filmatrix.for **
*****
** Tema : Transbasa los valores del vector Y a la matriz var **
** al igual que sus derivadas **
** Descripcion: **
*****

SUBROUTINE Filmatrix(j,y,dy)

INCLUDE 'StirSimple.inc'

INTEGER i,j

REAL*8 y(ROWV)
REAL*8 dy(ROWD)

DO i=1,ROWV
var(i,j) = y(i)
END DO

DO i=1,ROWD
dvar(i,j) = dy(i)
END DO

END SUBROUTINE Filmatrix

*****
** Modelo Simple Análisis de un motor Stirling **
*****
** Tema : Simulación de un motor Stirling mediante un modelo **
** adiabático ideal **
** Descripcion: Se grafican temperatura y energía vs theta **
*****
** Subroutine : Adiabatic.for **
** Programa Principal : StirSimple.for **
** Autor : Gustavo Scarpin **
** Fecha : 3 de Junio de 2012 **
** Ultima modificación: 3 de Junio de 2012 **
*****
```



```
SUBROUTINE Adiabatic

    INCLUDE 'StirSimple.inc'

C Definir las variables locales
    REAL*8 Qkpower, Qrpower, Qhpower, Wpower, eff

C Hacer el análisis adiabático ideal
    CALL adiab

C Imprimir los resultados del análisis adiabático ideal
    Qkpower = var(QK,COL)*freq      !Calor transferido al enfriador [W]
    Qrpower = var(QR,COL)*freq      !Calor transferido al regenerador [W]
    Qhpower = var(QH,COL)*freq      !Calor transferido al calentador [W]
    Wpower  = var(W ,COL)*freq      !Potencia total de salida [W]
    eff     = var(W,COL)/var(QH,COL) !Eficiencia térmica del motor [-]

    WRITE(*,1000) Qkpower, Qrpower, Qhpower, Wpower, eff*100.DO

C Archivo de salida para graficar los resultados de la simulación adiabática ideal
    CALL Plotadiab

1000 FORMAT('----- Analisis Adiabatico Ideal -----'/
.2X,'Calor transferido al enfriador [W]: ',F10.4/
.2X,'Calor transferido al regenerador [W]: ',F10.4/
.2X,'Calor transferido al calentador [W]: ',F10.4/
.2X,'Potencia total de salida [W]: ',F10.4/
.2X,'Eficiencia termica del motor [%]: ',F10.4/
'-----')

END SUBROUTINE Adiabatic

*****
**          Modelo Simple Análisis de un motor Stirling          **
*****
** Tema      : Genera un archivo para los gráficos pv y p-theta del **
**            análisis de Schmidt                                **
** Descripcion: Con formato para ser leído fácilmente desde Excel **
*****
** Subroutine : Plotpv.for                                       **
** Programa Principal : StirSimple.for                          **
** Autor      : Gustavo Scarpin                                  **
** Fecha     : 2 de Junio de 2012                               **
** Última modificación: 2 de Junio de 2012                     **
*****

SUBROUTINE Plotadiab

    INCLUDE 'StirSimple.inc'

C Definir las variables locales
    INTEGER i,j
    REAL*8 theta

    OPEN(UNIT=4,FILE='PlotAdiab.grf',STATUS='UNKNOWN')

    WRITE(4,1000)
    j=0
    DO theta=0.DO,360.DO,10.DO
        j = j + 1
        WRITE(4,'(50E15.5)') theta, (var(i,j),i=1,ROWV)
    END DO

1000 FORMAT(T3,' Theta [gr]          Tc [K]          Te [K]          Qk [J]
.          Qr [K]          Qh [K]          Wc [J]          We [J]
.          W [J]          P [Pa]          Vc [m3]          Ve [m3]          Mc [
.kg]          Mk [kg]          Mr [kg]          Mh [kg]          Me [kg]
.          Tck [K]          The [K]          GACK [kg/rad] GAKr [kg/rad] GARh
.[kg/rad] GAhe [kg/rad]')
    CLOSE(4)

END SUBROUTINE Plotadiab
```



```
*****
**                               Modelo Simple Análisis de un motor Stirling          **
*****
** Tema       : Geometría y propiedades térmicas del tubo conexión                **
**           : con el enfriador                                                  **
** Descripcion: Se especifica la geometría tipo tubo unitario                    **
*****
** Subroutine   : Conk.for                                                       **
** Programa Principal : StirSimple.for                                           **
** Autor       : Gustavo Scarpin                                                **
** Fecha      : 28 de Julio de 2012                                             **
** Última modificación: 28 de Julio de 2012                                     **
*****
```

SUBROUTINE Conk

INCLUDE 'StirSimple.inc'

NAMELIST/Geom_Conk/dck,lck

C Lee por archivo la geometria del regenerador

READ(1,Geom_Conk)

C Cálculo de las propiedades geométricas del regenerador

awgck = pi*dck*lck
ack = pi*dck*dck/4.D0
vclc = ack*lck+vclc0

WRITE(*,Geom_Conk)

END SUBROUTINE Conk

```
*****
**                               Modelo Simple Análisis de un motor Stirling          **
*****
** Tema       : Geometría y propiedades térmicas del tubo conexión                **
**           : con el calentador                                                **
** Descripcion: Se especifica la geometría tipo tubo unitario                    **
*****
** Subroutine   : Conh.for                                                       **
** Programa Principal : StirSimple.for                                           **
** Autor       : Gustavo Scarpin                                                **
** Fecha      : 28 de Julio de 2012                                             **
** Última modificación: 28 de Julio de 2012                                     **
*****
```

SUBROUTINE Conh

INCLUDE 'StirSimple.inc'

NAMELIST/Geom_Conh/dch,lch

C Lee por archivo la geometria del regenerador

READ(1,Geom_Conh)

C Cálculo de las propiedades geométricas del regenerador

awgch = pi*dch*lch
ach = pi*dch*dch/4.D0
vcle = ach*lch+vcle0

WRITE(*,Geom_Conh)

END SUBROUTINE Conh

```
*****
**                               Modelo Simple Análisis de un motor Stirling          **
*****
** Tema       : Evalua las derivadas del modelo adiabático ideal                **
** Descripcion:                                                                    **
*****
```



```
** Subroutine          : Dadiab.for          **
** Programa Principal : StirSimple.for      **
** Autor              : Gustavo Scarpin    **
** Fecha              : 3 de Junio de 2012 **
** Última modificación: 3 de Junio de 2012 **
*****
```

```
SUBROUTINE Dadiab(theta,y,dy)
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definir las variables locales
```

```
REAL*8 vot, top, bottom, dpop, theta
```

```
REAL*8 y(ROWV)
```

```
REAL*8 dy(ROWD)
```

```
C Volúmenes y sus derivadas
```

```
CALL volume(theta,y(VC),y(VE),dy(VC),dy(VE))
```

```
C Presión y sus derivadas
```

```
vot = vk/tk + vr/tr + vh/th
```

```
y(P) = mgas*rgas/(y(VC)/y(TC) + vot + y(VE)/y(TE))
```

```
top = -y(P)*(dy(VC)/y(TCK) + dy(VE)/y(THE))
```

```
bottom = (y(VC)/(y(TCK)*gama) + vot + y(VE)/(y(TE)*gama))
```

```
dy(P) = top/bottom
```

```
C Masas y sus derivadas (caudal másico)
```

```
y(MC) = y(P)*y(VC)/(rgas*y(TC))
```

```
y(MK) = y(P)*vk/(rgas*tk)
```

```
y(MR) = y(P)*vr/(rgas*tr)
```

```
y(MH) = y(P)*vh/(rgas*th)
```

```
y(ME) = y(P)*y(VE)/(rgas*y(TE))
```

```
dpop = dy(P)/y(P)
```

```
dy(MC) = (y(P)*dy(VC) + y(VC)*dy(P)/gama)/(rgas*y(TCK))
```

```
dy(MK) = y(MK)*dpop
```

```
dy(MR) = y(MR)*dpop
```

```
dy(MH) = y(MH)*dpop
```

```
dy(ME) = (y(P)*dy(VE) + y(VE)*dy(P)/gama)/(rgas*y(TE))
```

```
C Flujo másico entre celdas
```

```
y(GACK) = -dy(MC)
```

```
y(GAKR) = y(GACK) - dy(MK)
```

```
y(GAHE) = dy(ME)
```

```
y(GARH) = y(GAHE) + dy(MH)
```

```
C Condicional de temperatura entre celdas
```

```
y(TCK) = tk
```

```
IF (y(GACK).GT.0.D0) y(TCK) = y(TC)
```

```
y(THE) = y(TE)
```

```
IF (y(GAHE).GT.0.D0) y(THE) = th
```

```
C Las 7 derivadas a ser integradas mediante rk4
```

```
C Temperaturas de las zonas donde se produce trabajo
```

```
dy(TC) = y(TC)*(dpop + dy(VC)/y(VC) - dy(MC)/y(MC))
```

```
dy(TE) = y(TE)*(dpop + dy(VE)/y(VE) - dy(ME)/y(ME))
```

```
C Energia
```

```
dy(QK) = vk*dy(P)*cv/rgas - cp*(y(TCK)*y(GACK) - tk*y(GAKR))
```

```
dy(QR) = vr*dy(P)*cv/rgas - cp*(tk*y(GAKR) - th*y(GARH))
```

```
dy(QH) = vh*dy(P)*cv/rgas - cp*(th*y(GARH) - y(THE)*y(GAHE))
```

```
dy(WC) = y(P)*dy(VC)
```

```
dy(WE) = y(P)*dy(VE)
```

```
C Trabajo neto realizado
```

```
dy(W) = dy(WC) + dy(WE)
```

```
y(W) = y(WC) + y(WE)
```

```
END SUBROUTINE Dadiab
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling **
*****
** Tema                          : Define los parámetros y variables de entrada **
*****
```



```
** Descripcion: Define los parámetros geométricos y operacionales **
** del motor Stirling. Entrada sólo por archivo. **
*****
** Subroutine : Define.for **
** Programa Principal : StirSimple.for **
** Autor : Gustavo Scarpin **
** Fecha : 31 de Mayo de 2012 **
** Ultima modificación: 31 de Mayo de 2012 **
*****
```

```
SUBROUTINE Define
OPEN(UNIT=1,FILE='StirSimple.dat',STATUS='OLD')
CALL Engine
CALL Heatex
CALL Gas
CALL Operat
CLOSE(1)
END SUBROUTINE Define
```

```
*****
** Subroutine: Operat.for **
*****
** Tema : Determina los parámetros de operacion **
** Descripcion: **
*****
```

```
SUBROUTINE Operat
INCLUDE 'StirSimple.inc'
NAMELIST/Operacion/pmean,tk,th,freq
NAMELIST/Datos_Operacion/pmean,tk,th,tr,freq
READ(1,Operacion)
tr = (th - tk)/DLOG(th/tk)
omega = 2.D0*pi*freq
WRITE(*,Datos_Operacion)
END SUBROUTINE Operat
```

```
*****
** Modelo Simple Análisis de un motor Stirling **
*****
** Tema : Define la configuración y parámetros geométricos del **
** motor **
** Descripcion: Se prevé un sólo tipo de cinemática: senoidal **
*****
** Subroutine : Engine.for **
** Programa Principal : StirSimple.for **
** Autor : Gustavo Scarpin **
** Fecha : 1 de Junio de 2012 **
** Ultima modificación: 1 de Junio de 2012 **
*****
```

```
SUBROUTINE Engine
CALL Sindrive
END SUBROUTINE Engine
```

```
*****
** Subroutine: Sindrive.for **
*****
** Tema : Configuración de motor con cinemática senoidal **
** Descripcion: Definición de la geometría **
*****
```

```
SUBROUTINE Sindrive
```



```
INCLUDE 'StirSimple.inc'

NAMELIST/Motor/vswc,vswc,vclc0,vcle0,phase

READ(1,Motor)
WRITE(*,Motor)

alpha = phase * grad_a_rad

END SUBROUTINE Sindrive
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema       : Evalua la matriz del regenerador tipo alambre                                     **
** Descripción: Calcula el número de Stanton, factor de fricción de Stanton, factor de fricción de **
**              Reynolds y coeficiente de transferencia de calor                               **
*****
** Subroutine   : FrAlambre.for                                                                 **
** Programa Principal : StirSimple.for                                                         **
** Autor        : Gustavo Scarpin                                                             **
** Fecha        : 26 de Junio de 2012                                                         **
** Última modificación: 26 de Junio de 2012                                                 **
*****
```

```
SUBROUTINE FrAlambre(d,mu,re,st,ht,fr)
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definir las variables locales
```

```
C ENTRADAS
```

```
REAL*8 d      !Diámetro hidráulico [m]
REAL*8 mu     !Viscosidad dinámica del gas [kg*m/s]
REAL*8 re     !Número de Reynolds
```

```
C SALIDAS
```

```
REAL*8 st     !Número de Stanton
REAL*8 ht     !Coeficiente de transferencia de calor [W/(K*m**2)]
REAL*8 fr     !Factor de fricción de Reynolds Cref (= re*Factor de Fricción de Fanning)
              !Es igual a f*re del White dividido 4: fr=f*re/4.
```

```
IF (re.LT.940.D0) THEN      !Normalmente flujo laminar
    fr = 13.325D0
ELSE
    fr = 0.0791D0*re**0.75D0
END IF
```

```
C De la analogía simple de Reynolds
```

```
st = fr/(2.D0*re*prandtl)
ht = st*re*cp*mu/d
```

```
END SUBROUTINE FrAlambre
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema       : Evalua la matriz del regenerador tipo lámina                                     **
** Descripción: Calcula el número de Stanton, factor de fricción de Stanton, factor de fricción de **
**              Reynolds y coeficiente de transferencia de calor                               **
*****
** Subroutine   : FrFoil.for                                                                 **
** Programa Principal : StirSimple.for                                                         **
** Autor        : Gustavo Scarpin                                                             **
** Fecha        : 3 de Junio de 2012                                                         **
** Última modificación: 3 de Junio de 2012                                                 **
*****
```

```
SUBROUTINE FrFoil(d,mu,re,st,ht,fr)
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definir las variables locales
```

```
C ENTRADAS
```



```
REAL*8 d      !Diámetro hidráulico [m]
REAL*8 mu     !Viscosidad dinámica del gas [kg*m/s]
REAL*8 re     !Número de Reynolds
C SALIDAS
REAL*8 st     !Número de Stanton
REAL*8 ht     !Coeficiente de transferencia de calor [W/(K*m**2)]
REAL*8 fr     !Factor de fricción de Reynolds Cref (= re*Factor de Fricción de Fanning)
              !Es igual a f*re del White dividido 4: fr=f*re/4.

IF (re.LT.2000.D0) THEN    !Normalmente flujo laminar
  fr = 24.D0
ELSE
  fr = 0.0791D0*re**0.75D0
END IF

C De la analogía simple de Reynolds
st = fr/(2.D0*re*prandtl)
ht = st*re*cp*mu/d

END SUBROUTINE FrFoil
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema       : Evalua la matriz del regenerador tipo malla                                         **
** Descripcion: Calcula el número de Stanton y factor de fricción                                   **
*****
** Subroutine   : FrMatrix.for                                                                    **
** Programa Principal : StirSimple.for                                                            **
** Autor       : Gustavo Scarpin                                                                  **
** Fecha      : 3 de Junio de 2012                                                                **
** Última modificación: 3 de Junio de 2012                                                       **
*****
```

```
SUBROUTINE FrMatrix(re,st,fr)
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definir las variables locales
```

```
C ENTRADAS
```

```
REAL*8 re      !Número de Reynolds
```

```
C SALIDAS
```

```
REAL*8 st      !Número de Stanton
```

```
REAL*8 fr      !Factor de fricción de Reynolds Cref (= re*Factor de Fricción de Fanning)
```

```
              !Es igual a f*re del White dividido 4: fr=f*re/4.
```

```
C Ecuaciones tomadas de Kays & London (1955 edition)
```

```
st = 0.46D0*re**(-0.4D0)/prandtl
```

```
fr = 54.D0 + 1.43D0*re**0.78D0
```

```
END SUBROUTINE FrMatrix
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema       : Evalua el coeficiente de transf. de calor y factor                                   **
** de fricción de Reynolds                                                                       **
** Descripcion:                                                                                   **
*****
** Subroutine   : FrMatrix.for                                                                    **
** Programa Principal : StirSimple.for                                                            **
** Autor       : Gustavo Scarpin                                                                  **
** Fecha      : 3 de Junio de 2012                                                                **
** Última modificación: 3 de Junio de 2012                                                       **
*****
```

```
SUBROUTINE FrPipe(d,mu,re,ht,fr)
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definir las variables locales
```

```
C ENTRADAS
```



```
REAL*8 d      !Diámetro hidráulico [m]
REAL*8 mu     !Viscosidad dinámica del gas [kg*m/s]
REAL*8 re     !Número de Reynolds
C SALIDAS
REAL*8 ht     !Coeficiente de transferencia de calor [W/(K*m**2)]
REAL*8 fr     !Factor de fricción de Reynolds Cref (= re*Factor de Fricción de Fanning)
              !Es igual a f*re del White dividido 4: fr=f*re/4.
```

C Debido a las oscilaciones del flujo se asume que el mismo es siempre turbulento.

C Se usará la relación de Blasius para todos los números de Reynolds

```
fr = 0.0791D0*re**0.75D0
```

C De la analogía simple de Reynolds

```
ht = fr*mu*cp/(2.D0*d*prandtl)
```

```
END SUBROUTINE FrPipe
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema      : Especifica las propiedades del gas de trabajo                                         **
** Descripción: Se especifican los siguientes gases: Air, He, H2                                     **
**           Para el rango de los gases de trabajo usados en los motores Stirling y para el rango de temperatura de **
**           interés (entre 300 y 1000 K), el Número de Prandtl es aproximadamente constante alrededor de 0.7 **
*****
** Subroutine   : Gas.for                                                                              **
** Programa Principal : StirSimple.for                                                                **
** Autor       : Gustavo Scarpin                                                                     **
** Fecha      : 2 de Junio de 2012                                                                    **
** Última modificación: 2 de Junio de 2012                                                            **
*****
```

```
SUBROUTINE Gas
```

```
INCLUDE 'StirSimple.inc'
```

```
NAMELIST/Tipo_Gas/t_gas
```

```
NAMELIST/Datos_Gas/t_gas,gama,rgas,mu0,t0,t_suth,cp,cv,prandtl
```

```
READ(1,Tipo_Gas)
```

C INGRESA LAS PROPIEDADES DEL GAS DE TRABAJO

C ai)re he)lio hi)drógeno

```
IF (t_gas.eq.'ai') THEN
```

```
CALL Air
```

```
ELSE IF (t_gas.eq.'he') THEN
```

```
CALL Helium
```

```
ELSE IF (t_gas.eq.'hi') THEN
```

```
CALL Hydrogen
```

```
ELSE
```

```
STOP 'TIPO DE GAS NO DEFINIDO'
```

```
END IF
```

```
cv = rgas/(gama - 1.D0)
```

```
cp = gama*cv
```

```
t0 = 273.D0
```

```
prandtl = 0.71D0 !
```

```
WRITE(*,Datos_Gas)
```

```
END SUBROUTINE Gas
```



```
*****
** Subroutine:           Air.for                               **
*****
** Tema      : Datos termodinámicos del Aire                **
** Descripcion: Obtenidos del programa de Urieli            **
*****
```

```
SUBROUTINE Air

INCLUDE 'StirSimple.inc'

gama = 1.40D0
rgas = 287.00D0
mu0 = 17.08D-6
t_suth = 112.00D0

END SUBROUTINE Air
```

```
*****
** Subroutine:           Helium.for                           **
*****
** Tema      : Datos termodinámicos del Helio                **
** Descripcion: Obtenidos del programa de Urieli            **
*****
```

```
SUBROUTINE Helium

INCLUDE 'StirSimple.inc'

gama = 1.67D0
rgas = 2078.60D0
mu0 = 18.85D-6
t_suth = 80.00D0

END SUBROUTINE Helium
```

```
*****
** Subroutine:           Hydrogen.for                         **
*****
** Tema      : Datos termodinámicos del Hidrógeno            **
** Descripcion: Obtenidos del programa de Urieli            **
*****
```

```
SUBROUTINE Hydrogen

INCLUDE 'StirSimple.inc'

gama = 1.40D0
rgas = 4157.20D0
mu0 = 8.35D-6
t_suth = 84.40D0

END SUBROUTINE Hydrogen
```

```
*****
** Modelo Simple Análisis de un motor Stirling                **
*****
** Tema      : Geometría de los intercambiadores de calor    **
** Descripcion: Especifica la geometría de tres tipos de intercamb. **
*****
** Subroutine      : Heatex.for                                **
** Programa Principal : StirSimple.for                        **
** Autor          : Gustavo Scarpin                          **
** Fecha         : 2 de Junio de 2012                        **
** Última modificación: 2 de Junio de 2012                  **
*****
```

```
SUBROUTINE Heatex

INCLUDE 'StirSimple.inc'

NAMELIST/Inter/t_enfriador,t_calentador,t_regenerador,
.
t_matriz
```



```
READ(1,Inter)
```

```
CALL Cooler  
CALL Conk  
CALL Regen  
CALL Conh  
CALL Heater
```

```
END SUBROUTINE Heatex
```

```
*****  
** Subroutine:          Cooler.for          **  
*****  
** Tema                : Especifica los parámetros geométricos del cooler    **  
** Descripción: Definición de la geometría del enfriador                    **  
*****
```

```
SUBROUTINE Cooler
```

```
INCLUDE 'StirSimple.inc'
```

```
NAMELIST/Enfriador/t_enfriador,vk,ak,awgk,dk,lk
```

```
C INGRESA LA GEOMETRIA SEGUN EL TIPO DE ENFRIADOR  
C t)ubos a)nillo r)anurado m)alla de alambres
```

```
IF (t_enfriador.EQ.'t') THEN
```

```
    CALL Pipes('k',vk,ak,awgk,dk,lk)          !Enfriador formado por tubos
```

```
ELSE IF (t_enfriador.EQ.'a') THEN
```

```
    CALL Annulus('k',vk,ak,awgk,dk,lk)      !Enfriador tipo anillo
```

```
ELSE IF (t_enfriador.EQ.'r') THEN
```

```
    CALL Slots('k',vk,ak,awgk,dk,lk)       !Enfriador ranurado
```

```
ELSE IF (t_enfriador.EQ.'m') THEN
```

```
    CALL Tela('k',vk,ak,awgk,dk,lk)       !Enfriador tipo malla de alambres
```

```
ELSE
```

```
    STOP 'TIPO DE ENFRIADOR NO DEFINIDO'
```

```
END IF
```

```
WRITE(*,Enfriador)
```

```
END SUBROUTINE Cooler
```

```
*****  
** Subroutine:          Heater.for          **  
*****  
** Tema                : Especifica los parámetros geométricos del heater    **  
** Descripción: Definición de la geometría del calentador                    **  
*****
```

```
SUBROUTINE Heater
```

```
INCLUDE 'StirSimple.inc'
```

```
NAMELIST/Calentador/t_calentador,vh,ah,awgh,dh,lh
```

```
C INGRESA LA GEOMETRIA SEGUN EL TIPO DE CALENTADOR  
C t)ubos a)nillo r)anurado m)alla de alambres
```

```
IF (t_calentador.EQ.'t') THEN
```

```
    CALL Pipes('h',vh,ah,awgh,dh,lh)       !Calentador formado por tubos
```



```
ELSE IF (t_calentador.EQ.'a') THEN
    CALL Annulus('h',vh,ah,awgh,dh,lh) !Calentador tipo anillo
ELSE IF (t_calentador.EQ.'r') THEN
    CALL Slots('h',vh,ah,awgh,dh,lh) !Calentador ranurado
ELSE IF (t_calentador.EQ.'m') THEN
    CALL Tela('h',vh,ah,awgh,dh,lh) !Calentador tipo malla de alambres
ELSE
    STOP 'TIPO DE CALENTADOR NO DEFINIDO'
END IF
WRITE(*,Calentador)
END SUBROUTINE Heater
```

```
*****
** Subroutine: Pipes.for **
*****
** Tema : Geometria de un intercambiador formado por caños **
** Descripcion: Se consideran caños lisos **
*****
```

```
SUBROUTINE Pipes(tipo,v,a,awg,d,len)
```

```
C Definir las variables locales
```

```
CHARACTER*1 tipo !Tipo de intercambiador de calor
INTEGER num !Número de caños del paquete
REAL*8 v !Volumen libre del intercambiador de calor [m^3]
REAL*8 a !Area de flujo libre interna del intercambiador de calor [m^2]
REAL*8 awg !Area mojada interna del intercambiador de calor [m^2]
REAL*8 d !Diámetro interno del tubo [m]
REAL*8 len !Longitud del intercambiador de calor [m]
```

```
REAL*8, PARAMETER :: PI = 3.14159265359D0
```

```
NAMELIST/Geom_Enf/d,len,num
```

```
NAMELIST/Geom_Cal/d,len,num
```

```
C Lee por archivo la geometria del intercambiador de calor
```

```
IF (tipo.EQ.'k') THEN
    READ(1,Geom_Enf)
ELSE IF (tipo.EQ.'h') THEN
    READ(1,Geom_Cal)
ELSE
    STOP 'Error en la subrutina Pipes'
END IF
```

```
C Cálculo de las propiedades geométricas del intercambiador de calor
```

```
a = num*pi*d*d/4.D0
v = a*len
awg = num*pi*d*len
```

```
END SUBROUTINE Pipes
```

```
*****
** Subroutine: Annulus.for **
*****
** Tema : Geometria de un intercambiador anular **
** Descripcion: Se consideran un sólo anillo **
*****
```

```
SUBROUTINE Annulus(tipo,v,a,awg,d,len)
```

```
C Definir las variables locales
```

```
CHARACTER*1 tipo !Tipo de intercambiador de calor
REAL*8 dout !Diámetro externo de la ranura del anillo [m]
REAL*8 din !Diámetro interno de la ranura del anillo [m]
```



```
REAL*8 v          !Volumen libre del intercambiador de calor [m^3]
REAL*8 a          !Area de flujo libre interna del intercambiador de calor [m^2]
REAL*8 awg        !Area mojada interna del intercambiador de calor [m^2]
REAL*8 d          !Diámetro hidráulico [m]
REAL*8 len        !Longitud del intercambiador de calor [m]
```

```
REAL*8, PARAMETER :: PI = 3.14159265359D0
```

```
NAMELIST/Geom_Enf/dout,din,len
NAMELIST/Geom_Cal/dout,din,len
```

C Lee por archivo la geometria del intercambiador de calor

```
IF (tipo.EQ.'k') THEN
  READ(1,Geom_Enf)
ELSE IF (tipo.EQ.'h') THEN
  READ(1,Geom_Cal)
ELSE
  STOP 'Error en la subrutina Annulus'
END IF
```

C Cálculo de las propiedades geométricas del intercambiador de calor

```
a = pi*(dout*dout - din*din)/4.D0
v = a*len
awg = pi*dout*len
d = dout - din
```

```
END SUBROUTINE Annulus
```

```
*****
** Subroutine:          Slots.for          **
*****
** Tema                : Geometria de un intercambiador ranurado          **
** Descripcion         :                                                  **
*****
```

```
SUBROUTINE Slots(tipo,v,a,awg,d,len)
```

C Definir las variables locales

```
CHARACTER*1 tipo  !Tipo de intercambiador de calor
INTEGER num      !Número de ranuras del paquete
REAL*8 w         !Ancho de la ranura [m]
REAL*8 h         !Alto de la ranura [m]
REAL*8 v         !Volumen libre del intercambiador de calor [m^3]
REAL*8 a         !Area de flujo libre interna del intercambiador de calor [m^2]
REAL*8 awg       !Area mojada interna del intercambiador de calor [m^2]
REAL*8 d         !Diámetro hidráulico [m]
REAL*8 len       !Longitud del intercambiador de calor [m]
```

```
REAL*8, PARAMETER :: PI = 3.14159265359D0
```

```
NAMELIST/Geom_Enf/w,h,len,num
NAMELIST/Geom_Cal/w,h,len,num
```

C Lee por archivo la geometria del intercambiador de calor

```
IF (tipo.EQ.'k') THEN
  READ(1,Geom_Enf)
ELSE IF (tipo.EQ.'h') THEN
  READ(1,Geom_Cal)
ELSE
  STOP 'Error en la subrutina Slots'
END IF
```

C Cálculo de las propiedades geométricas del intercambiador de calor

```
a = num*w*h
v = a*len
awg = num*(w + 2.D0*h)*len
d = 4.D0*v/awg
```

```
END SUBROUTINE Slots
```

```
*****
** Subroutine:          Tela.for          **
*****
```



```
*****
** Tema      : Geometria de un intercambiador formado por telas      **
**           : metlalicas                                           **
** Descripcion:                                                       **
*****
```

```
      SUBROUTINE Tela(tipo,v,a,awg,d,len)
```

```
      C Definir las variables locales
```

```
      CHARACTER*1 tipo      !Tipo de intercambiador de calor
      REAL*8 v              !Volumen libre del intercambiador de calor [m^3]
      REAL*8 a              !Area de flujo libre interna del intercambiador de calor [m^2]
      REAL*8 awg            !Area mojada interna del intercambiador de calor [m^2]
      REAL*8 d              !Diámetro hidráulico[m]
      REAL*8 len            !Longitud del intercambiador de calor [m]
      REAL*8 porosity       !Porosidad de la tela de alambres
      REAL*8 din            !Diámetro interno del tubo [m]
      REAL*8 dwire          !Diámetro de los alambres de la tela [m]
```

```
      REAL*8, PARAMETER :: PI = 3.14159265359D0
```

```
      NAMELIST/Geom_Enf/din,len,dwire,porosity
      NAMELIST/Geom_Cal/din,len,dwire,porosity
```

```
      C Lee por archivo la geometria del intercambiador de calor
```

```
      IF (tipo.EQ.'k') THEN
         READ(1,Geom_Enf)
```

```
         write(*,*) 'Catidad de mallas del enfriador=',len/2.D0/dwire
         pause
```

```
      ELSE IF (tipo.EQ.'h') THEN
         READ(1,Geom_Cal)
```

```
         write(*,*) 'Catidad de mallas del calentador=',len/2.D0/dwire
         pause
```

```
      ELSE
         STOP 'Error en la subrutina Tela'
      END IF
```

```
      C Cálculo de las propiedades geométricas del intercambiador de calor
```

```
         awgr0 = pi*din*len
```

```
         a = pi*din*din/4.D0*porosity
         v = a*len
         d = dwire*porosity/(1.D0 - porosity)
         awg = 4.D0*v/d + awgr0
```

```
      END SUBROUTINE Tela
```

```
*****
**           Modelo Simple Análisis de un motor Stirling           **
*****
** Tema      : Genera un archivo para los gráficos pv y p-theta del **
**           : análisis de Schmidt                               **
** Descripcion: Con formato para ser leído fácilmente desde Excel **
*****
** Subroutine      : Plotpv.for                                  **
** Programa Principal : StirSimple.for                          **
** Autor          : Gustavo Scarpin                             **
** Fecha         : 2 de Junio de 2012                           **
** Última modificación: 2 de Junio de 2012                       **
*****
```

```
      SUBROUTINE Plotpv
```

```
         INCLUDE 'StirSimple.inc'
```

```
      C Definir las variables locales
```

```
      REAL*8 theta,plot_vc,plot_ve,plot_p,plot_vtot
```

```
      OPEN(UNIT=4,FILE='SchmidtPV.grf',STATUS='UNKNOWN')
```



```
WRITE(4,1000)

DO theta=0.D0,360.D0,5.D0
  plot_vc= vclc + 0.5D0*vswc*(1.D0 + DCOS(theta*grad_a_rad))
  plot_ve= vcle + 0.5D0*vswe*(1.D0 + DCOS(theta*grad_a_rad+alpha))
  plot_p = mgas*rgas/(plot_vc/tk +vk/tk+vr/tr+vh/th + plot_ve/th)
  plot_vtot = (plot_vc + vk + vr + vh + plot_ve)

  WRITE(4,'(10F15.5)') theta,plot_vc*1.D+6,plot_ve*1.D+6,
  . plot_vtot*1.D+6,plot_p*1.D-5,pmean*1.D-5
END DO

1000 FORMAT('      Theta [gr]      Vc [cm^3]      Ve [cm^3]      Vtot [cm^
.3]      P [Bar]      Pmean [Bar]')

CLOSE(4)
END SUBROUTINE Plotpv

*****
** Subroutine:          Plotmass.for          **
*****
** Tema                : Genera un archivo para los gráficos de          **
**                    Mapa de Trayectoria de Partículas                **
** Descripción: Con formato para ser leído fácilmente desde Excel        **
*****

SUBROUTINE Plotmass

INCLUDE 'StirSimple.inc'

C Definir las variables locales
REAL*8 NT                !Relación de temperatura
REAL*8 Vref              !Volumen de referencia
REAL*8 vswe_r            !Volumen de barrido de expansión reducido
REAL*8 vcle_r            !Volumen libre de compresión reducido
REAL*8 vh_r              !Volumen libre del calentador reducido
REAL*8 vr_r              !Volumen libre del regenerador reducido
REAL*8 vk_r              !Volumen libre del enfriador reducido
REAL*8 vclc_r            !Volumen libre de compresión reducido
REAL*8 vswc_r            !Volumen de barrido de compresión reducido
REAL*8 L1                !Límite del volumen de barrido de expansión reducido
REAL*8 L2                !Límite del volumen libre de expansión reducido
REAL*8 L3                !Límite del volumen libre del calentador reducido
REAL*8 L4                !Límite del volumen libre del regenerador reducido
REAL*8 L5                !Límite del volumen libre del enfriador reducido
REAL*8 L6                !Límite del volumen de barrido de compresión reducido
REAL*8 theta,ve_r(100),vc_r,vt_r(100),plot_v(50),Ve_e,vt_rmin
INTEGER i,j,m,n,step

OPEN(UNIT=4,FILE='SchmidtMass.grf',STATUS='UNKNOWN')

NT = th/tk
Vref = vswe

vswe_r = (vswe/Vref)/NT
vcle_r = (vcle/Vref)/NT
vh_r = (vh/Vref)/NT
vr_r = (vr/Vref)*DLOG(NT)/(NT-1.D0)
vk_r = (vk/Vref)
vswc_r = (vswc/Vref)
vclc_r = (vclc/Vref)

vt_rmin = 1.D10
i = 0
DO theta=0.D0,360.D0,5.D0
  i = i + 1

  Ve_e = (vswe/2)*(1.D0-DCOS(theta*grad_a_rad))          !Volumen de expansion vs fase
  Vc_r = (vswc/2)*(1.D0+DCOS(theta*grad_a_rad-alpha))    !Volumen de compresión vs fase
  ve_r(i) = (Ve_e/Vref)/NT !Volumen de expansión reducido vs fase
  vc_r = Vc_r/Vref          !Volumen de compresión reducido vs fase
  vt_r(i) = vswe_r + vcle_r + vh_r + vr_r + vk_r + vclc_r + vc_r !Volumen total vs fase

  IF (vt_r(i).LE.vt_rmin) vt_rmin = vt_r(i)
```



```
END DO

n=i

L1 = vswe_r
L2 = L1 + vcle_r
L3 = L2 + vh_r
L4 = L3 + vr_r
L5 = L4 + vk_r
L6 = vt_rmin

step = 30
i = 0
DO theta=0.D0,360.D0,5.D0
  i = i + 1
  DO m=0,step

    plot_v(m+1) = ve_r(i) + (REAL(m)/REAL(step))*(vt_r(i) - ve_r(i)) !Segmentos de volúmenes
reducidos

  END DO
  WRITE(4,'(50F15.5)') theta,L1,L2,L3,L4,L5,L6,
. (plot_v(j+1),j=0,step)
END DO

CLOSE(4)
END SUBROUTINE Plotmass
```

```
*****
** Subroutine:          Plotloss.for          **
*****
** Tema                : Genera un archivo para los gráficos de          **
**                    : pérdida de presión y calor                      **
** Descripción:       : Con formato para ser leído fácilmente desde Excel **
**                    : Llamado desde SimWork.for                       **
*****
```

```
SUBROUTINE Plotloss(dpck,dpkol,dpreg,dphot,dpch,dp,pcom,pexp)

IMPLICIT NONE

INTEGER i
REAL*8 re(100), dp(100), dpkol(100), dpreg(100), dphot(100)
REAL*8 dpck(100), dpch(100)
REAL*8 pcom(100), pexp(100), theta

OPEN(UNIT=4,FILE='PlotLoss.grf',STATUS='UNKNOWN')

WRITE(4,1000)

i = 0
DO theta=0.D0,360.D0,10.D0

  i=i+1

  WRITE(4,'(10F15.5)') theta,dpck(i),dpkol(i),dpreg(i),dphot(i),
. dpch(i),dp(i),pcom(i)*1.D-5,pexp(i)*1.D-5

END DO

1000 FORMAT('      Theta [gr]      dPck [Pa]      dPkol [Pa]      dPreg [P
.a]      dPphot [Pa]      dPch [Pa]      dPtot [Pa]      Padiab [Bar] Ps
.imple [Bar]')

CLOSE(4)

END SUBROUTINE Plotloss
```

```
*****
**                      Modelo Simple Análisis de un motor Stirling      **
*****
```



```
** Tema      : Geometría y propiedades térmicas del regenerador      **
** Descripcion: Se especifica la geometría de dos tipos de          **
** regeneradores: tubulares y anulares                             **
*****
** Subroutine   : Regen.for                                          **
** Programa Principal : StirSimple.for                              **
** Autor        : Gustavo Scarpin                                   **
** Fecha        : 2 de Junio de 2012                               **
** Última modificación: 2 de Junio de 2012                         **
*****
```

```
SUBROUTINE Regen
```

```
INCLUDE 'StirSimple.inc'
```

```
NAMELIST/Regenerador/t_regenerador,vr,ar,awgr,dr,lr,awgr0,cqwr,
.
kwr,amat,awr,porosity
```

```
C INGRESA LA GEOMETRIA SEGUN EL TIPO DE REGENERADOR
C t)ubular a)nular
```

```
IF (t_regenerador.EQ.'t') THEN
```

```
CALL Tubular                                !Regenerador tubular
```

```
ELSE IF (t_regenerador.EQ.'a') THEN
```

```
CALL Annular                                !Regenerador anular
```

```
ELSE
```

```
STOP 'TIPO DE REGENERADOR NO DEFINIDO'
```

```
END IF
```

```
CALL Matrix                                !Propiedades de la matriz
```

```
CALL Ther_Cond                              !Cálculo de la conductividad térmica
```

```
WRITE(*,Regenerador)
```

```
END SUBROUTINE Regen
```

```
*****
** Subroutine:      Tubular.for                                          **
*****
** Tema      : Geometria del regenerador formado por tubos          **
** Descripcion:                                             **
*****
```

```
SUBROUTINE Tubular
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definir las variables locales
```

```
INTEGER num                                !Número de tubos
```

```
REAL*8 dout                                !Diámetro externo del carter del tubo [m]
```

```
REAL*8 domat                                !Diámetro interno del carter del tubo [m]
```

```
NAMELIST/Geom_Reg/dout,domat,lr,num
```

```
C Lee por archivo la geometria del regenerador
```

```
READ(1,Geom_Reg)
```

```
C Cálculo de las propiedades geométricas del regenerador
```

```
awgr0 = num*pi*domat*lr
```

```
amat = num*pi*domat*domat/4.D0
```

```
awr = num*pi*(dout*dout - domat*domat)/4.D0
```

```
END SUBROUTINE Tubular
```

```
*****
```



```
** Subroutine:          Annular.for          **
*****
** Tema              : Geometria del regenerador de tipo anular  **
** Descripcion:      Se define un sólo anillo                    **
*****

SUBROUTINE Annular

INCLUDE 'StirSimple.inc'

C Definir las variables locales
REAL*8 dout          !Diámetro externo del carter del tubo [m]
REAL*8 domat         !Diámetro interno del carter del tubo [m]
REAL*8 dimat         !Diámetro interno de la matriz [m]

NAMELIST/Geom_Reg/dout,domat,dimat,lr

C Lee por archivo la geometria del regenerador

READ(1,Geom_Reg)

C Cálculo de las propiedades geométricas del regenerador

awgr0 = pi*(dimat + domat)*lr
amat = pi*(domat*domat - dimat*dimat)/4.D0
awr = pi*(dout*dout - domat*domat)/4.D0

END SUBROUTINE Annular

*****
** Subroutine:          Matrix.for          **
*****
** Tema              : Especifica la geometría de la matriz del regenerador **
** Descripcion:      Se define tres tipos de matrices                **
*****

SUBROUTINE Matrix

INCLUDE 'StirSimple.inc'

C INGRESA LA GEOMETRIA SEGUN EL TIPO DE MATRIZ DEL REGENERADOR
C m)alla de alambres l)lámina a)lambres apilados s)in matriz

IF (t_matriz.EQ.'m') THEN

CALL Mesh                      !Matriz tipo malla de alambres

ELSE IF (t_matriz.EQ.'l') THEN

CALL Foil                      !Matriz tipo lámina metálica plegada

ELSE IF (t_matriz.EQ.'a') THEN

CALL Alambre                   !Matriz tipo alambres apilados

ELSE IF (t_matriz.EQ.'s') THEN

CALL NoMatrix                  !Sin matriz

ELSE

STOP 'TIPO DE MATRIZ NO DEFINIDA'

END IF

END SUBROUTINE Matrix

*****
** Subroutine:          Mesh.for           **
*****
** Tema              : Geometria del regenerador con matriz tipo  **
**                   malla de alambres                             **
** Descripcion:      Se define un sólo anillo                    **
*****

SUBROUTINE Mesh
```



```
INCLUDE 'StirSimple.inc'

C Definir las variables locales
REAL*8 dwire      !Diámetro de los alambres [m]

NAMELIST/Geom_Malla/porosity,dwire

C Lee por archivo la geometria del regenerador

READ(1,Geom_Malla)

C Cálculo de las propiedades geométricas del regenerador

ar   = amat*porosity
vr   = ar*lr
dr   = dwire*porosity/(1.D0 - porosity)
awgr = 4.D0*vr/dr + awgr0

write(*,*) 'Catidad de mallas del regenerador=',lr/2.D0/dwire
pause

END SUBROUTINE Mesh

*****
** Subroutine:                Foil.for                **
*****
** Tema          : Geometria del regenerador con matriz tipo **
**               : lámina metálica plegada             **
** Descripcion:                **
*****

SUBROUTINE Foil

INCLUDE 'StirSimple.inc'

C Definir las variables locales
REAL*8 fl      !Longitud de la lámina metálica sin plegar [m]
REAL*8 tf      !Espesor de la lámina metálica [m]
REAL*8 am      !Area de la lámina [m^2]

NAMELIST/Geom_Malla/fl,tf

C Lee por archivo la geometria del regenerador

READ(1,Geom_Malla)

C Cálculo de las propiedades geométricas del regenerador

am   = tf*fl
ar   = amat - am
vr   = ar*lr
awgr = 2.D0*lr*fl + awgr0
dr   = 4.D0*vr/awgr
porosity = ar/amat

END SUBROUTINE Foil

*****
** Subroutine:                Alambre.for            **
*****
** Tema          : Geometria del regenerador con matriz tipo **
**               : alambres apilados                 **
** Descripcion:                **
*****

SUBROUTINE Alambre

INCLUDE 'StirSimple.inc'

C Definir las variables locales
REAL*8 dal      !Diámetro del alambre [m]
REAL*8 nal      !Real indicando el número de alambre aproximados

NAMELIST/Geom_Alambre/dal
```



C Lee por archivo la geometria del regenerador

```
READ(1,Geom_Alambre)
```

C Cálculo de las propiedades geométricas del regenerador

```
nal = amat/(DSQRT(3.D0)*dal*dal/2.D0)
ar = (DSQRT(3.D0)-PI/2.D0)*dal*dal/2.D0*nal
vr = ar*lr
awgr = PI*dal*lr*nal + awgr0
dr = DSQRT((DSQRT(3.D0)-PI/2)*2.D0/PI)*dal
porosity = 1.D0-PI/(2.D0*DSQRT(3.D0))
```

```
END SUBROUTINE Alambre
```

```
*****
** Subroutine:          NoMatrix.for          **
*****
** Tema      : Geometria del regenerador sin matriz **
** Descripcion: **
*****
```

```
SUBROUTINE NoMatrix
```

```
INCLUDE 'StirSimple.inc'
```

C Cálculo de las propiedades geométricas del regenerador

```
ar = amat
vr = ar*lr
awgr = awgr0
dr = 4.D0*vr/awgr
porosity = 1.D0
```

```
END SUBROUTINE NoMatrix
```

```
*****
** Subroutine:          Ther_Cond.for          **
*****
** Tema      : Conductividad térmica del carter del regenerador **
** Descripcion: Se asume un carter fabricado de acero inoxidable. **
**           A pesar de que depende de la temperatura, se asume **
**           un valor constante de conductividad termica: **
**           25 [W/m/K] para condiciones normales **
*****
```

```
SUBROUTINE Ther_Cond
```

```
INCLUDE 'StirSimple.inc'
```

C Cálculo de las propiedades geométricas del regenerador

```
c   kwr = 25.D0      !Conductividad térmica del Acero [W/m/K]
    kwr = 0.3D0      !Conductividad térmica del Caucho Vulcanizado [W/m/K]
c   kwr = 0.03D0     !Conductividad térmica del Aerogel de sílice [W/m/K]
    cqwr = kwr*awr/lr !Conductividad térmica de la pared del carter del regenerador [W/K]
```

```
END SUBROUTINE Ther_Cond
```

```
*****
** Modelo Simple Análisis de un motor Stirling **
*****
** Tema      : Evalua la viscosidad dinámica, conductividad termica **
**           y número de Reynolds **
** Descripcion: Longitud de referencia: diámetro hidráulico **
*****
** Subroutine      : Reynum.for **
** Programa Principal : StirSimple.for **
** Autor          : Gustavo Scarpin **
** Fecha         : 3 de Junio de 2012 **
** Ultima modificación: 3 de Junio de 2012 **
*****
```

```
SUBROUTINE Reynum(t,gmas,d,mu,kgas,re)
```



```
INCLUDE 'StirSimple.inc'

C Definir las variables locales
C ENTRADAS
REAL*8 t      !Temperatura del gas [K]
REAL*8 gmas   !Flujo másico [kg/(s*m^2)]
REAL*8 d      !Diámetro hidráulico [m]
C SALIDAS
REAL*8 mu     !Viscosidad dinámica del gas [kg/(m*s)]
REAL*8 kgas   !Conductividad térmica del gas [W/(m*K)]
REAL*8 re     !Número de Reynolds [-]

mu = mu0*(t0 + t_suth)/(t + t_suth)*(t/t0)**1.5D0
kgas = cp*mu/prandtl
re = DABS(gmas)*d/mu

IF (re.LT.1.D0) re = 1.D0

END SUBROUTINE Reynum
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema       : Metodo clasico de integracion numerica de ecuaciones **
**             diferenciales, Runge-Kutta de cuarto orden           **
** Descripcion: Integra n ecuaciones diferenciales                  **
**             Se fija la subrutina Dadiab                          **
*****
** Subroutine   : RK4.for                                           **
** Programa Principal : StirSimple.for                             **
** Autor        : Gustavo Scarpin                                  **
** Fecha       : 3 de Junio de 2012                               **
** Ultima modificación: 3 de Junio de 2012                       **
*****
```

```
SUBROUTINE RK4(n,x,dx,y,dy,Ny,Ndy)

INTEGER i, n, Ny, Ndy
REAL*8 x,dx,y(Ny),dy(Ndy),dy1(Ndy),dy2(Ndy),dy3(Ndy),x0,y0(Ny),xm

x0 = x
y0 = y

CALL Dadiab(x0,y,dy1)
DO i=1,n
  y(i) = y0(i) + 0.5D0*dx*dy1(i)
END DO

xm = x0 + 0.5D0*dx
CALL Dadiab(xm,y,dy2)
DO i=1,n
  y(i) = y0(i) + 0.5D0*dx*dy2(i)
END DO

CALL Dadiab(xm,y,dy3)
DO i=1,n
  y(i) = y0(i) + dx*dy3(i)
END DO

x = x0 + dx
CALL Dadiab(x,y,dy)

DO i=1,n
  dy(i) = (dy1(i) + 2.D0*(dy2(i) + dy3(i)) +dy(i))/6.D0
  y(i) = y0(i) + dx*dy(i)
END DO

END SUBROUTINE RK4
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
```



```
** Tema      : Realiza el análisis de Schmidt          **
** Descripc: Incluye salida de datos para los gráficos **
**          - p vs v del análisis de Schmidt          **
**          - p vs theta del análisis de Schmidt      **
**          - distribución de partículas de masas de Alan Organ **
*****
** Programa Principal : StirSimple.for                **
** Autor              : Gustavo Scarpin              **
** Fecha              : 2 de junio de 2012           **
** Última modificación: 2 de junio de 2012         **
*****
```

```
SUBROUTINE Schmidt
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definición de variables locales
```

```
REAL*8 c,s,b,sqrtb,bf,work,power,eff,qc,qe
```

```
NAMELIST/A_Schmidt/beta,mgas
```

```
C Análisis de Schmidt
```

```
c = DSQRT((vswe/th)**2.D0 + (vswc/tk)**2.D0 + 2.D0*(vswe/th)*
. (vswc/tk)*DCOS(alpha))/2.D0
s = (vswc/2.D0 + vc1c +vk)/tk + vr/tr +(vswe/2.D0 + vc1e + vh)/th
b = c/s
sqrtb = DSQRT(1.D0 - b**2.D0)
bf = (1.D0 - 1.D0/sqrtb)
beta = DATAN(vswe*DSIN(alpha)/th/(vswe*DCOS(alpha)/th + vswc/tk))
mvol = s*sqrtb !Volumen promedio del motor [m^3]
mgas = pmean*mvol/rgas !Masa total de gas en el motor [kg]
```

```
C Trabajo producido
```

```
qc = pi*vswc*mgas*rgas*DSIN(beta)*bf/c
qe = pi*vswe*mgas*rgas*DSIN(beta - alpha)*bf/c
work = (qc + qe)
power = work*freq
eff = work/qe*100.D0
```

```
C Resultados del Análisis de Schmidt
```

```
WRITE(*,A_Schmidt)
```

```
WRITE(*,1000) work,power,qe,qc,eff
```

```
CALL plotpv
```

```
CALL plotmass
```

```
1000 FORMAT('----- Analisis de Schmidt -----'/
.1X,'Trabajo [joule]',F10.4,3X,'Potencia [watts]',F10.4/
.1X,'Qexpans [joule]',F10.4,3X,'Qcompres [joule]',F10.4/
.11X,'Eficiencia Indicada [%]',F10.4/
. '-----')
```

```
END SUBROUTINE Schmidt
```

```
*****
** Modelo Simple Análisis de un motor Stirling          **
*****
** Tema      : Evalua la performance de transf. de calor promedio **
**          del calentador                                **
** Descripc:                                          **
*****
** Subroutine   : SimHot.for                            **
** Programa Principal : StirSimple.for                  **
** Autor        : Gustavo Scarpin                      **
** Fecha        : 3 de Junio de 2012                   **
** Última modificación: 3 de Junio de 2012             **
*****
```

```
SUBROUTINE SimHot(twh,qgross,tgh)
```

```
INCLUDE 'StirSimple.inc'
```



```
C Definir las variables locales
C ENTRADAS
  REAL*8 twh      !Temperatura de la pared del calentador [K]
  REAL*8 qrloss  !Pérdida de calor debido al regenerador imperfecto [J]
C SALIDAS
  REAL*8 tgh      !Temperatura promedio del gas del calentador

  INTEGER i
  REAL*8 gah(100), gh, sumre, remax, reavg, mu, kgas, re(100), ht,fr

C Número de Reynolds sobre el ciclo
DO i=1,37
  gah(i) = (var(GARH,i) + var(GAHE,i))*omega/2.DO
  gh      = gah(i)/ah
  CALL Reynum(th,gh,dh,mu,kgas,re(i))
END DO

C Número de Reynolds promedio y máximo
sumre = 0.DO
remax = re(1)
DO i=1,36
  sumre = sumre + re(i)
  IF (re(i).GT.remmax) remax = re(i)
END DO

reavg = sumre/36.DO

CALL FrPipe(dh,mu,reavg,ht,fr)  !Coeficiente de transferencia de calor

tgh = twh - (var(QH,37)+qrloss)*freq/(ht*awgh) !Temperatura del gas en el calentador [K]

WRITE(*,1000) reavg, remax, ht, twh, tgh

1000 FORMAT('----- Analisis Simple del Calentador -----'/
.2X,'Numero de Reynolds promedio      : ',F10.4/
.2X,'Numero de Reynolds maximo       : ',F10.4/
.2X,'Coef. de transf. de calor [W/K/m^2] : ',F10.4/
.2X,'Temp. de pared del calentador [K] : ',F10.4/
.2X,'Temp. del gas en el calentador [K] : ',F10.4/
.'-----')

END SUBROUTINE SimHot
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema      : Evalua la performance de transf. de calor promedio                               **
**                               del enfriador                                                       **
** Descripcion:                                                                                       **
*****
** Subroutine   : SimKol.for                               **
** Programa Principal : StirSimple.for                     **
** Autor        : Gustavo Scarpin                          **
** Fecha        : 3 de Junio de 2012                       **
** Última modificación: 3 de Junio de 2012                 **
*****
```

```
SUBROUTINE SimKol(twk,qrloss,tgk)
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definir las variables locales
C ENTRADAS
  REAL*8 twk      !Temperatura de la pared del enfriador [K]
  REAL*8 qrloss  !Pérdida de calor debido al regenerador imperfecto [J]
C SALIDAS
  REAL*8 tgk      !Temperatura promedio del gas del enfriador

  INTEGER i
  REAL*8 gak(100), gk, sumre, remax, reavg, mu, kgas, re(100), ht,fr

C Número de Reynolds sobre el ciclo
DO i=1,37
  gak(i) = (var(GACK,i) + var(GAKR,i))*omega/2.DO
  gk      = gak(i)/ak
```



```
CALL Reynum(tk,gk,dk,mu,kgas,re(i))
END DO

C Número de Reynolds promedio y máximo
sumre = 0.D0
remax = re(1)
DO i=1,36
  sumre = sumre + re(i)
  IF (re(i).GT.remax) remax = re(i)
END DO

reavg = sumre/36.D0

CALL FrPipe(dk,mu,reavg,ht,fr) !Coeficiente de transferencia de calor

tgk = twk - (var(QK,37)-qrloss)*freq/(ht*awgk) !Temperatura del gas en el enfriador [K]

WRITE(*,1000) reavg, remax, ht, twk, tgk

1000 FORMAT('----- Analisis Simple del Enfriador -----'/
.2X,'Numero de Reynolds promedio : ',F10.4/
.2X,'Numero de Reynolds maximo : ',F10.4/
.2X,'Coef. de transf. de calor [W/K/m^2] : ',F10.4/
.2X,'Temp. de pared del enfriador [K] : ',F10.4/
.2X,'Temp. del gas en el enfriador [K] : ',F10.4/
.'-----')

END SUBROUTINE SimKol
```

```
*****
** Modelo Simple Análisis de un motor Stirling **
*****
** Tema : Modelo adiabático ideal con efectos de calor y pres. **
** Descripción: Incluye los efectos de la transferencia de calor y **
** pérdida de presión **
*****
** Subroutine : Simple.for **
** Programa Principal : StirSimple.for **
** Autor : Gustavo Scarpin **
** Fecha : 3 de Junio de 2012 **
** Última modificación: 3 de Junio de 2012 **
*****
```

```
SUBROUTINE Simple
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definir las variables locales
REAL*8, PARAMETER :: epsilon = 1.D0 !Error de temperatura admisible [K]
REAL*8 twk, twh, tgh, tgk, terror, qrloss, dwork
REAL*8 Qkpower, Qrpower, Qhpower, Wpower, eff, qwrl
REAL*8 actWpower !Potencia real de salida del motor [W]
REAL*8 actQhpower !Calor real transf. al calentador [W]
REAL*8 acteff !Eficiencia termica real del motor[-]

C Condiciones iniciales
twk = tk !Temp. de la pared del enfriador = Temp. inicial del gas en el enfriador
twh = th !Temp. de la pared del calentador= Temp. inicial del gas en el calentador

terror = 10.D0*epsilon !Error inicial a la entrada del ciclo de convergencia

C Ciclo de convergencia
DO WHILE (terror.GE.epsilon)
  CALL Adiab
  CALL SimReg(qrloss) !Entalpia perdida en el regenerador [J]
  CALL SimHot(twh,qrloss,tgh) !Nueva temperatura de gas del calentador [K]
  CALL SimKol(twk,qrloss,tgk) !Nueva temperatura de gas del enfriador [K]
  terror = DABS(th - tgh) + DABS(tk - tgk)
  th = tgh
  tk = tgk
  tr = (th-tk)/DLOG(th/tk)
END DO

C Imprimir los resultados del análisis adiabático ideal
Qkpower = var(QK,COL)*freq !Calor transferido al enfriador [W]
```



```
Qrpower = var(QR,COL)*freq      !Calor transferido al regenerador [W]
Qhpower = var(QH,COL)*freq      !Calor transferido al calentador [W]
Wpower  = var(W ,COL)*freq      !Potencia total de salida [W]
eff      = var(W,COL)/var(QH,COL) !Eficiencia térmica del motor [-]

WRITE(*,1000) Qkpower, Qrpower, Qhpower, Wpower, eff*100.D0

C Archivo de salida para graficar los resultados de la simulación adiabática ideal
CALL Plotadiab

C Imprimir los resultados del análisis del regenerador
qwrl = cqwr*(twh - twk)/freq

WRITE(*,1100) qrloss*freq, qwrl*freq

C Imprimir los resultados del análisis de la pérdida de presión
CALL SimWork(dwork) !Evalúa la pérdida de trabajo disponible debido a la pérdida de presión

actWpower = Wpower - dwork*freq
actQhpower = Qhpower + qrloss*freq + qwrl*freq
acteff = actWpower/actQhpower

WRITE(*,1200) dwork*freq, actQhpower, actWpower, acteff*100.D0

1000 FORMAT('----- Analisis Adiabatico Ideal Simple -----'/
.2X,'Calor transferido al enfriador [W]: ',F10.4/
.2X,'Calor transferido al regenerador [W]: ',F10.4/
.2X,'Calor transferido al calentador [W]: ',F10.4/
.2X,'Potencia total de salida [W]: ',F10.4/
.2X,'Eficiencia termica del motor [%]: ',F10.4/
.'-----')

1100 FORMAT('----- Analisis de Resultados del Regenerador -----'/
.2X,'Entalpia neta perdida [W]: ',F10.4/
.2X,'Fuga de calor en la pared [W]: ',F10.4/
.'-----')

1200 FORMAT('----- Analisis Simple por Perdidas -----'/
.2X,'Trab.perdido por perdida de Pres.[W]: ',F10.4/
.2X,'Calor real transf. al calentador [W]: ',F10.4/
.2X,'Potencia real [W]: ',F10.4/
.2X,'Eficiencia termica real del motor[%]: ',F10.4/
.'-----')

END SUBROUTINE Simple
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema                          : Evalua la eficiencia y performance del regenerador                       **
** Descripcion: Calcula la pérdida de entalpía del regenerador:qrloss**
*****
** Subroutine                    : SimReg.for                                                            **
** Programa Principal            : StirSimple.for                                                         **
** Autor                         : Gustavo Scarpin                                                       **
** Fecha                         : 3 de Junio de 2012                                                    **
** Ultima modificación: 3 de Junio de 2012                                                              **
*****

SUBROUTINE SimReg(qrloss)

INCLUDE 'StirSimple.inc'

C Definir las variables locales
INTEGER i
REAL*8 gr, gar(100), re(100), qrloss, sumre, remax, reavg, ntu, st
REAL*8 fr, ht, effect, qreg(100), qrmin, qrmax, mu, kgas

C Número de Reynolds sobre el ciclo
DO i=1,37
gar(i) = (var(GAKR,i) + var(GARH,i))*omega/2.D0
gr      = gar(i)/ar
CALL Reynum(tr,gr,dr,mu,kgas,re(i))
```



```
END DO

C Número de Reynolds promedio y máximo
sumre = 0.D0
remaX = re(1)
DO i=1,36
  sumre = sumre + re(i)
  IF (re(i).GT.remaX) remaX = re(i)
END DO

reavg = sumre/36.D0

C Número de Stanton, Número de unidades de transferencia, Efectividad del regenerador
IF (t_matriz.EQ.'m') THEN
  CALL FrMatrix(reavg,st,fr)
ELSE IF (t_matriz.EQ.'l') THEN
  CALL FrFoil(dr,mu,reavg,st,ht,fr)
ELSE IF (t_matriz.EQ.'a') THEN
  CALL FrAlambre(dr,mu,reavg,st,ht,fr)
ELSE IF (t_matriz.EQ.'s') THEN
  CALL FrFoil(dr,mu,reavg,st,ht,fr)
END IF

ntu = st*awgr/(2.D0*ar)
effect = ntu/(ntu + 1.D0)
write(*,*) 're',awgr,st,ar,ntu,effect

qrmin = var(QR,1)
qrmax = var(QR,1)
DO i=1,37
  qreg(i) = var(QR,i)
  IF (qreg(i).LT.qrmin) qrmin = qreg(i)
  IF (qreg(i).GT.qrmax) qrmax = qreg(i)
END DO

C Calcula la pérdida de entalpía del regenerador:qrloss
qrloss = (1.D0 - effect)*(qrmax - qrmin)

WRITE(*,1000) reavg, remaX, st, ntu, effect*100.D0, qrloss*freq

1000 FORMAT('----- Analisis Simple del Regenerador -----'/
.2X,'Numero de Reynolds promedio      : ',F10.4/
.2X,'Numero de Reynolds maximo        : ',F10.4/
.2X,'Numero de Stanton (a Re promedio) : ',F10.4/
.2X,'Numero de unidades de transferencia : ',F10.4/
.2X,'Eficiencia del regenerador        [%]: ',F10.4/
.2X,'Pérdida de Entalpía neta del reg.[W]: ',F10.4/
'-----')

END SUBROUTINE SimReg
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema      : Evalúa la pérdida de trabajo disponible debido a la pérdida de presión **
** Descripción: **
*****
** Subroutine   : SimWork.for **
** Programa Principal : StirSimple.for **
** Autor       : Gustavo Scarpin **
** Fecha      : 4 de Junio de 2012 **
** Última modificación: 4 de Junio de 2012 **
*****

SUBROUTINE SimWork(dwork)

INCLUDE 'StirSimple.inc'

C Definir las variables locales
REAL*8, PARAMETER :: dtheta = 2.D0*pi/36.D0

INTEGER i
REAL*8 dwork, gck, gk, gr, gh, gch, mu, kgas, st, fr, ht
REAL*8 re(100), dp(100), dpkol(100), dpreg(100), dphot(100)
```



```
REAL*8 dpck(100), dpch(100), MasaCK, MasaCH
REAL*8 pcom(100), pexp(100)
```

```
C Evaluación de la pérdida de presión
dwork = 0.D0
```

```
DO i=1,36
```

```
C Pérdida de presión en el tubo de conexión con el enfriador
gck = var(GACK,i)*omega/ack
```

```
CALL Reynum(tk,gck,dck,mu,kgas,re(i))
```

```
CALL FrPipe(dck,mu,re(i),ht,fr)
MasaCK = var(P,i)*vclc/(rgas*tk)
```

```
dpck(i) = 2.D0*fr*mu*vclc*gck*lck/(MasaCK*dck**2.D0)
```

```
C Pérdida de presión en el enfriador
gk = (var(GACK,i) + var(GAKR,i))*omega/(2.D0*ak)
```

```
CALL Reynum(tk,gk,dk,mu,kgas,re(i))
```

```
IF (t_enfriador.EQ.'m') THEN
  CALL FrMatrix(re(i),st,fr) !Matriz tipo malla de alambres
ELSE
  CALL FrPipe(dk,mu,re(i),ht,fr) !Sin matriz
ENDIF
```

```
dpkol(i) = 2.D0*fr*mu*vk*gk*lk/(var(MK,i)*dk**2.D0)
```

```
C Pérdida de presión en el regenerador
gr = (var(GAKR,i) + var(GARH,i))*omega/(2.D0*ar)
```

```
CALL Reynum(tr,gr,dr,mu,kgas,re(i))
```

```
IF (t_matriz.EQ.'m') THEN
  CALL FrMatrix(re(i),st,fr) !Matriz tipo malla de alambres
ELSE IF (t_matriz.EQ.'l') THEN
  CALL FrFoil(dr,mu,re(i),st,ht,fr) !Matriz tipo lámina metálica plegada
ELSE IF (t_matriz.EQ.'a') THEN
  CALL FrAlambre(dr,mu,re(i),st,ht,fr) !Matriz tipo alambres apilados
ELSE IF (t_matriz.EQ.'s') THEN
  CALL FrPipe(dr,mu,re(i),ht,fr) !Sin matriz
END IF
```

```
dpreg(i) = 2.D0*fr*mu*vr*gr*lr/(var(MR,i)*dr**2.D0)
```

```
C Pérdida de presión en el calentador
gh = (var(GARH,i) + var(GAHE,i))*omega/(2.D0*ah)
```

```
CALL Reynum(th,gh,dh,mu,kgas,re(i))
```

```
IF (t_calentador.EQ.'m') THEN
  CALL FrMatrix(re(i),st,fr) !Matriz tipo malla de alambres
ELSE
  CALL FrPipe(dh,mu,re(i),ht,fr) !Sin Matriz
ENDIF
```

```
dphot(i) = 2.D0*fr*mu*vh*gh*lh/(var(MH,i)*dh**2.D0)
```

```
C Pérdida de presión en el tubo de conexión con el enfriador
gch = var(GAHE,i)*omega/ach
```

```
CALL Reynum(th,gch,dch,mu,kgas,re(i))
```

```
CALL FrPipe(dch,mu,re(i),ht,fr)
MasaCH = var(P,i)*vcle/(rgas*th)
```

```
dpch(i) = 2.D0*fr*mu*vcle*gch*lch/(MasaCH*dch**2.D0)
```

```
C Pérdida de presión total [Pa]
dp(i) = dpck(i) + dpkol(i) + dpreg(i) + dphot(i) + dpch(i)
```



```
C Trabajo perdido debido a la pérdida de presión [J]
  dwork = dwork + dtheta*dp(i)*dvar(VE,i)

  pcom(i) = var(P,i)
  pexp(i) = pcom(i) + dp(i)

END DO

dpck(COL) = dpck(1)
dpkol(COL) = dpkol(1)
dpreg(COL) = dpreg(1)
dphot(COL) = dphot(1)
dpch(COL) = dpch(1)
dp(COL) = dp(1)
pcom(COL) = pcom(1)
pexp(COL) = pexp(1)

CALL Plotloss(dpck,dpkol,dpreg,dphot,dpch,dp,pcom,pexp)

END SUBROUTINE SimWork
```

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
** Tema       : Determina las variaciones de volumen en las zonas de trabajo                               **
** Descripción: Se preve sólo movimiento sinusoidal, motor Alpha                                       **
*****
** Subroutine   : Volume.for                                                                                   **
** Programa Principal : StirSimple.for                                                                     **
** Autor        : Gustavo Scarpin                                                                           **
** Fecha        : 3 de Junio de 2012                                                                       **
** Última modificación: 3 de Junio de 2012                                                                 **
*****
```

```
SUBROUTINE Volume(theta,Volc,Vole,dVolc,dVole)
```

```
IMPLICIT NONE
```

```
C Definir las variables locales
REAL*8 theta,Volc,Vole,dVolc,dVole

CALL Sinevol(theta,Volc,Vole,dVolc,dVole)

END SUBROUTINE Volume
```

```
*****
** Subroutine: Sinevol.for                                                                                   **
*****
** Tema       : Variaciones de volumen y sus derivadas para un movimiento sinusoidal                       **
** Descripción: Los datos geométricos fueron ingresados en Sindrive                                       **
*****
```

```
SUBROUTINE Sinevol(theta,Volc,Vole,dVolc,dVole)
```

```
INCLUDE 'StirSimple.inc'
```

```
C Definir las variables locales (volumenes y sus derivadas)
REAL*8 theta,Volc,Vole,dVolc,dVole

Volc = vclc + 0.5D0*vswc*(1.D0 + DCOS(theta + pi))
Vole = vcle + 0.5D0*vswe*(1.D0 + DCOS(theta + pi + alpha))
dVolc = -0.5D0*vswc*DSIN(theta + pi)
dVole = -0.5D0*vswe*DSIN(theta + pi + alpha)

END SUBROUTINE Sinevol
```

INCLUDE

```
*****
**                               Modelo Simple Análisis de un motor Stirling                               **
*****
```



```
*****
** Tema      : Definicion de variables                               **
** Descripcio: Definicion de variables globales                     **
*****
** Incluir   : StirSimple.inc                                       **
** Programa Principal : StirSimple.for                             **
** Autor     : Gustavo Scarpin                                       **
** Fecha    : 31 de Mayo de 2012                                     **
** Ultima modificación: 31 de Mayo de 2012                         **
*****

IMPLICIT NONE

C DEFINICION DE CONSTANTES GLOBALES
  REAL*8, PARAMETER :: PI = 3.14159265359D0
  REAL*8, PARAMETER :: grad_a_rad = 0.01745329252D0
  REAL*8, PARAMETER :: rad_a_grad = 57.2957795131D0

C INDICE DE LAS FILAS DE LOS ARREGLOS var Y dvar.
  INTEGER, PARAMETER :: TC = 1 !Temperatura de la zona de compresión [K]
  INTEGER, PARAMETER :: TE = 2 !Temperatura de la zona de expansión [K]
  INTEGER, PARAMETER :: QK = 3 !Calor transferido al enfriador [J]
  INTEGER, PARAMETER :: QR = 4 !Calor transferido al regenerador [J]
  INTEGER, PARAMETER :: QH = 5 !Calor transferido al calentador [J]
  INTEGER, PARAMETER :: WC = 6 !Trabajo realizado por la zona de compresión [J]
  INTEGER, PARAMETER :: WE = 7 !Trabajo realizado por la zona de expansión [J]
  INTEGER, PARAMETER :: W = 8 !Trabajo total realizado (WC + WE) [J]
  INTEGER, PARAMETER :: P = 9 !Presión [Pa]
  INTEGER, PARAMETER :: VC = 10 !Volumen de la zona de compresión [m^3]
  INTEGER, PARAMETER :: VE = 11 !Volumen de la zona de expansión [m^3]
  INTEGER, PARAMETER :: MC = 12 !Masa de gas en la zona de compresión [kg]
  INTEGER, PARAMETER :: MK = 13 !Masa de gas en el enfriador [kg]
  INTEGER, PARAMETER :: MR = 14 !Masa de gas en el regenerador [kg]
  INTEGER, PARAMETER :: MH = 15 !Masa de gas en el calentador [kg]
  INTEGER, PARAMETER :: ME = 16 !Masa de gas en la zona de expansión [kg]
  INTEGER, PARAMETER :: TCK = 17 !Temperatura condicional zona de compresión / enfriador [K]
  INTEGER, PARAMETER :: THE = 18 !Temperatura condicional calentador / zona de expansión [K]
  INTEGER, PARAMETER :: GACK = 19 !Flujo de masa condicional zona de compresión / enfriador [kg/rad]
  INTEGER, PARAMETER :: GAKR = 20 !Flujo de masa condicional enfriador / regenerador [kg/rad]
  INTEGER, PARAMETER :: GARH = 21 !Flujo de masa condicional regenerador / calentador [kg/rad]
  INTEGER, PARAMETER :: GAHE = 22 !Flujo de masa condicional calentador / zona de expansión [kg/rad]

C DIMENSIONAMIENTO DE LOS ARREGLOS var Y dvar
  INTEGER, PARAMETER :: ROWV = 22 !Número de filas de la matriz var
  INTEGER, PARAMETER :: ROWD = 16 !Número de filas de la matriz dvar
  INTEGER, PARAMETER :: COL = 37 !Número de columnas en las matrices (cada 10 grados)

  REAL*8 var(ROWV,COL)
  REAL*8 dvar(ROWD,COL)

  COMMON/estado/var,dvar

C PARAMETROS GEOMETRICOS DEL MOTOR
  REAL*8 vclc, vcle ! Volúmenes libres de compresión y expansión totales [m^3]
  REAL*8 vclc0, vcle0 ! Volúmenes libres de las zonas compresión y expansión unicamente [m^3]
  REAL*8 vswc, vswe ! Volúmenes de barrido de compresión y expansión [m^3]
  REAL*8 alpha ! Angulo de fase de avance de la zona de expansión [radianes]
  REAL*8 phase ! Angulo de fase de avance de la zona de expansión [grados]

C PARAMETROS GEOMETRICOS DEL ENFRIADOR
  CHARACTER*1 t_enfriador !t)ubos a)nillo r)anurado m)matriz
  REAL*8 vk ! Volumen libre del enfriador [m^3]
  REAL*8 ak ! Area de flujo libre interna del enfriador [m^2]
  REAL*8 awgk ! Area mojada interna del enfriador [m^2]
  REAL*8 dk ! Diámetro hidráulico del enfriador [m]
  REAL*8 lk ! Longitud efectiva del enfriador [m]

C PARAMETROS GEOMETRICOS DEL CALENTADOR
  CHARACTER*1 t_calentador !t)ubos a)nillo r)anurado m)matriz
  REAL*8 vh ! Volumen libre del calentador [m^3]
  REAL*8 ah ! Area de flujo libre interna del calentador [m^2]
  REAL*8 awgh ! Area mojada interna del calentador [m^2]
  REAL*8 dh ! Diámetro hidráulico del calentador [m]
  REAL*8 lh ! Longitud efectiva del calentador [m]
```



```
C PARAMETROS GEOMETRICOS DEL TUBO DE CONEXION DEL ENFRIADOR
  REAL*8 ack      ! Area de flujo libre interna del tubo del enfriador [m^2]
  REAL*8 awgck   ! Area mojada interna del tubo del enfriador [m^2]
  REAL*8 dck     ! Diámetro hidráulico del tubo del enfriador [m]
  REAL*8 lck     ! Longitud efectiva del tubo del enfriador [m]

C PARAMETROS GEOMETRICOS DEL TUBO DE CONEXION DEL CALENTADOR
  REAL*8 ach     ! Area de flujo libre interna del tubo del calentador [m^2]
  REAL*8 awgch  ! Area mojada interna del tubo del calentador [m^2]
  REAL*8 dch    ! Diámetro hidráulico del tubo del calentador [m]
  REAL*8 lch    ! Longitud efectiva del tubo del calentador [m]

C PARAMETROS GEOMETRICOS DEL REGENERADOR
  CHARACTER*1 t_regenerador !t)ubular a)nular
  CHARACTER*1 t_matriz     !m)alla l)ámina s)in matriz
  REAL*8 vr               ! Volumen libre del regenerador [m^3]
  REAL*8 ar               ! Area de flujo libre interna del regenerador [m^2]
  REAL*8 awgr            ! Area mojada interna del regenerador [m^2]
  REAL*8 dr              ! Diámetro hidráulico del regenerador [m]
  REAL*8 lr              ! Longitud efectiva del regenerador [m]
  REAL*8 awgr0           ! Area mojada del regenerador sin matriz [m^2]
  REAL*8 cqwr            ! Conductividad térmica del carter del regenerador [m^2]
  REAL*8 kwr             ! Conductividad térmica del regenerador [W/m/K]
  REAL*8 amat            ! Area de la matriz del regenerador [m^2]
  REAL*8 awr             ! Area de pared del carter del regenerador [m^2]
  REAL*8 porosity        ! Porosidad de la matriz del regenerador [-]

COMMON/geometria/vclc,vcle,vswc,vswe,alpha,phase,vk,ak,awgk,dk,lk,
.      vh,ah,awgh,dh,lh,vr,ar,awgr,dr,lr,awgr0,cqwr,kwr,
.      amat,awr,porosity,vclc0,vcle0,
.      ack,awgck,dck,lck,ach,awgch,dch,lch,
.      t_enfriador,t_calentador,t_regenerador,t_matriz

C PARAMETROS DEL GAS DE TRABAJO
  CHARACTER*2 t_gas      ! ai)re he)lio hi)drógeno
  REAL*8 rgas           ! Constante del gas [J/kg/K]
  REAL*8 cp             ! Calor específico a presión constante [J/kg/K]
  REAL*8 cv             ! Calor específico a volumen constante [J/kg/K]
  REAL*8 gama           ! Constante adiabática, relación cp/cv [-]
  REAL*8 mu0            ! Viscosidad dinámica a la temperatura de referencia t0 [kg*m/s]
  REAL*8 t0, t_suth     ! Temperatura de referencia [K], Constante de Sutherland [K]
  REAL*8 prandtl        ! Número de Prandtl

COMMON/dato_gas/rgas,cp,cv,gama,mu0,t0,t_suth,prandtl

C PARAMETROS DE OPERACION
  REAL*8 pmean          ! Presión (de carga) media [Pa]
  REAL*8 tk, tr, th     ! Temperaturas del enfriador, regenerador y calentador [K]
  REAL*8 freq, omega    ! Frecuencia del ciclo [Herz], [rad/seg]
  REAL*8 mgas           ! Masa total de gas en el motor [kg]
  REAL*8 mvol           ! Volumen promedio del motor [m^3]
  REAL*8 beta           ! Angulo de fase de la presión [rad]

COMMON/operation/pmean,tk,tr,th,freq,omega,mgas,mvol,beta
```