รายการอ้างอิง

ภาษาไทย

ภัทรพรรณ ประศาสน์สารกิจ. การถ่ายโอนความร้อน . กรุงเทพมหานคร :ภาควิชาเคมีเทคนิค คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2530.

มนตรี พิรุณเกษตร การถ่ายเทความร้อนฉบับเตรียมสอบและเสริมประสบการณ์.พิมพ์ครั้งที่ 1. กรุงเทพมหานคร:วิทยพัฒน์, 2541.

วิวัฒน์ ตัณฑะพานิชกุล. อุปกรณ์แลกเปลี่ยนความร้อนในอุตสาหกรรม.พิมพ์ครั้งที่1. กรุงเทพมหานคร:บริษัท ส.เอเชียเพรส, 2536.

สุธรรม ศรีเกษม, เมธินทร์ ทรงชัยกุล และ สง่า ศรีศุภปรีดา . Matlab เพื่อการแก้ปัญหาทางวิศวกรรม. สำนักพิมพ์มหาวิทยาลัยรังสิต, 2521.

ภาษาอังกฤษ

Bird, R.B., Stewart, W. E. and Lightfoot, E.N. Transport Phenomena. New York : Department of Chemical Engineering University of Wisconsin,1960.

Devois, J.F., Durastanti, J.F. and Martin, B. Numerical modelling of the spiral plate heat exchanger. Journal of Thetmal Analysis Vol. 44,1995.

Huang, Y. and Sciver, S.W. Forced flow He II heat exchangers. Cryogenics. Vol. 36,1996.

Nacati, M.O. Heat transfer A Basic Approach. New York: McGraw-Hill International Edition,1985.

Patankar, S.V. Numerical Heat Transfer and Fluid Flow. New York : Hemisphere Publishing , Taylor & Francis Group,1980.

Schroder, E. and Buhler, K. Three –dimensional convection in rectangular domains with horizontal throughflow. Int. J.I Heat Mass Transfer Vol.38 No.7,1995.

Spalding, D.B. and Taborek J. Heat Exchanger design handbook. New York: Hemisphere Publishing , 1983.

Versteeg, H.K. and Malalasekera W. An introduction to Computational Fluid Dynamics The Finite Volume Method. New York: John Wiley & Sons,1995.

ภาคผนวก

ภาคผนวก ก

เทคนิคการคำนวณแบบ Power Law

วิธีการ power law ใช้สำหรับการประมาณค่าปัญหาการกระจายสมการไฟไนต์ดิฟเฟอ-เรนซ์ที่มีทั้งเทอมของการนำความร้อนและเทอมของการพาความร้อน วิธีนี้เป็นของ Patankar (1980) จัดเป็นวิธีที่มีความยุ่งยากซับซ้อนพอสมควรแต่ให้ผลการคำนวณได้ถูกต้องและใกล้เคียงกับวิธี exponential (Patankar,1980)

จากสมการที่มีทั้งเทอมของการนำและการพา กำหนดให้ F คือค่าเทอมของการไหลหรือการพาความร้อน ส่วน D คือเทอมของการแพร่หรือการนำความร้อน โดย $F_e = \rho u|_e$ และ $D_e = \frac{\Gamma_e}{\delta z}$ (หัวข้อที่ 3.2) การกระจาย power law ที่ a_E เขียนได้ดังนี้

1. For $P_e<-10$

$$\frac{a_E}{D_e} = -P_e$$
$$a_E = -F_e$$

2. For $-10 \le P_e < 0$

$$\frac{a_E}{D_e} = (1. + 0.1P_e)^5 - P_e$$
$$a_E = D_e(1. + 0.1P_e)^5 - F_e$$

3. For $0 \le P_e < 10$

$$\frac{a_E}{D_e} = (1. - 0.1P_e)^5$$
$$a_E = D_e(1. - 0.1P_e)^5$$

4. For $P_e>10$

$$\frac{a_E}{D_e} = 0$$
$$a_E = 0$$

เขียนให้อยู่ในรูปทั่วไปได้

$$a_E = D_e \max\left[0,\left(1.-0.1\frac{|F_e|}{D_e}\right)^5\right] + \max(0,-F_e)$$

จากสมการรูปทั่วไปข้างต้นจะเห็นได้ว่าเทคนิควิธีการคำนวณแบบ power law เหมือนกับ upwind differencing เมื่อ $|P_e| > 10$ ก็คือไม่คิดผลกระทบจากการนำความร้อนที่น้อยกว่าการพาความร้อนถึง 10 เท่า ในขณะเดียวกันจะคำนวณผลกระทบของการนำความร้อนเมื่อการพาความร้อนมากกว่าค่าการนำความร้อนไม่ถึง 10 เท่า หรือ $|P_e| < 10$

ภาคผนวก ข

การแก้สมการเมทริกซ์แบบสามแนวทแยงมุม

ในกรณีสมการเมทริกซ์ที่เกิดขึ้นจากการใช้วิธีการทางตัวเลขเป็นเมทริกซ์ที่ไม่เต็มเช่น เมทริกซ์ทีเกิดขึ้นจากการเทคนิควิธีการไม่ชัดแจ้ง (implicit) นั้นจะทำให้เกิดเมทริกซ์ที่มีสัมประสิทธิ์ของตัวแปรในแต่ละสมการเฉพาะในแนวทแยงมุม และบนกับล่างแนวทแยงมุมเท่านั้น เมทริกซ์นี้มีชื่อเรียกว่า เมทริกซ์แบบสามแนวทแยงมุม (tri-diagonal matrix, TDMA) ส่วนที่จุดอื่นๆ ของเมทริกซ์จะมีค่าสัมประสิทธิ์เป็นศูนย์ทั้งสิ้น การแก้สมการเมทริกซ์นี้มีวิธีพิเศษเฉพาะ ซึ่งจะลดจำนวนครั้งในการคำนวณลง พิจารณาเมทริกซ์แบบสามแนวทแยงมุมต่อไปนี้ (Versteeg และ Malalasekera, 1995)

$$
\begin{aligned}
\phi_1 &= C_1 \quad \ldots\ldots 1a \\
-\beta_2\phi_1 + D_2\phi_2 - \alpha_2\phi_3 &= C_2 \quad \ldots\ldots 1b \\
-\beta_3\phi_2 + D_3\phi_3 - \alpha_3\phi_4 &= C_3 \quad \ldots\ldots 1c \\
-\beta_4\phi_3 + D_4\phi_4 - \alpha_4\phi_5 &= C_4 \quad \ldots\ldots 1d \\
\cdots\quad &= \quad \cdots \\
\beta_n\phi_{n-1} + D_n\phi_n - \alpha_n\phi_{n+1} &= C_n \quad \ldots\ldots 1n \\
\phi_{n+1} &= C_{n+1} \quad \ldots\ldots 1n+1
\end{aligned}
$$

จากค่าภาวะขอบเขต ทำให้ทราบค่า ϕ_1 และ ϕ_{n+1} เขียนรูปทั่วไปของแต่ละสมการได้ดังนี้

$$-\beta_j\phi_{j-1} + D_j\phi_j - \alpha_j\phi_{j+1} = C_j \quad \ldots\ldots 2$$

สมการที่ 1b-1n สามารถจัดรูปใหม่ได้ว่า

$$\phi_2 = \frac{\alpha_2}{D_2}\phi_3 + \frac{\beta_2}{D_2}\phi_1 + \frac{C_2}{D_2} \quad \ldots\ldots 3a$$

$$\phi_3 = \frac{\alpha_3}{D_3}\phi_4 + \frac{\beta_3}{D_3}\phi_2 + \frac{C_3}{D_3} \quad \ldots\ldots 3b$$

$$\phi_4 = \frac{\alpha_4}{D_4}\phi_5 + \frac{\beta_4}{D_4}\phi_3 + \frac{C_4}{D_4} \quad \ldots\ldots 3c$$

.....

$$\phi_n = \frac{\alpha_n}{D_n}\phi_{n+1} + \frac{\beta_n}{D_n}\phi_n + \frac{C_n}{D_n}$$

สมการเหล่านี้สามารถคำนวณหาคำตอบโดยใช้วิธีการลดเทอมไปข้างหน้า (forward elimination) และการแทนที่เทอมย้อนหลัง (back-substitution) เริ่มด้วยการแทนที่เทอม ϕ_2 ในสมการ 3b โดยการแทนที่ด้วยสมการ 3a จะได้

$$\phi_3 = \left(\frac{\alpha_3}{D_3 - \beta_3 \frac{\alpha_2}{D_2}} \right) \phi_4 + \left(\frac{\beta_3 \left(\frac{\beta_2}{D_2} \phi_1 + \frac{C_2}{D_2} \right) + C_3}{D_3 - \beta_3 \frac{\alpha_2}{D_2}} \right) \quad \text{.....4a}$$

กำหนดให้

$$A_2 = \frac{\alpha_2}{D_2}$$

$$C_2' = \frac{\beta_2}{D_2} \phi_1 + \frac{C_2}{D_2} \quad \text{.....4b}$$

ดังนั้นสมการที่ 7a สามารถเขียนใหม่ได้ว่า

$$\phi_3 = \left(\frac{\alpha_3}{D_3 - \beta_3 A_2} \right) \phi_4 + \left(\frac{\beta_3 C_2' + C_3}{D_3 - \beta_3 A_2} \right) \quad \text{.....4c}$$

กำหนดให้

$$A_3 = \frac{\alpha_3}{D_3 - \beta_3 A_2}$$

$$C_3' = \frac{\beta_3 C_2' + C_3}{D_3 - \beta_3 A_2}$$

สมการที่ 4c สามารถเขียนให้อยู่ในรูปอย่างง่าย ดังนี้

$$\phi_3 = A_3 \phi_4 + C_3' \quad \text{.....5}$$

และจากสมการที่ 5 นำไปแทนที่เทอม $\phi 3$ ในสมการ 3c ทำเช่นเดียวกันนี้กับสมการถัดไปจนถึงสมการสุดท้าย

สำหรับการแทนที่ย้อนกลับ เขียนให้อยู่เดียวกับสมการที่ 5 ดังนี้

$$\phi_j = A_j \phi_{j+1} + C_j' \quad \text{.....6a}$$

เมื่อ

$$A_j = \frac{\alpha_j}{D_j - \beta_j A_{j-1}} \quad \text{.....6b}$$

$$C_j' = \frac{\beta_j C_{j-1}' + C_j}{D_j - \beta_j A_{j-1}} \quad \text{.....6c}$$

เนื่องจากเรารู้ค่าภาวะขอบเขตที่ j=1 และ j=n+1 นำมาใช้หาค่า A และ C′

$$A_1 = 0, C_1' = \phi_1$$

$$A_{n+1} = 0, C_{n+1}' = \phi_{n+1}$$

จากค่า α_j, β_j, D_j และ C_j ในสมการที่ 2 สามารถนำมาใช้หาค่า A_j' และ C_j' ได้จากนั้นเริ่มคำนวณโดยแทนค่า A_j' และ C_j' ที่ j=2 ไปจนถึง j=n (6b-c) ดังนั้นถ้ารู้ค่า ϕ ที่ภาวะขอบเขตที่

ตำแหน่ง n+1 ค่า ϕj จะสามารถย้อนกลับมาคำนวณได้ตามลำดับ (ϕ_n, ϕ_{n-1},...., ϕ_2) จากสมการที่ 6a

เทคนิคการการใช้ TDMA ในการแก้ปัญหาการถ่ายเทความร้อน 2 มิติ

วิธีการ TDMA (tri-diagonal matrix algorithm) สามารถนำมาประยุกต์ใช้หาคำตอบปัญหาสมการ 2 มิติ ได้ จากรูปที่ 1 สมการรูปทั่วไปใน 2 มิติ

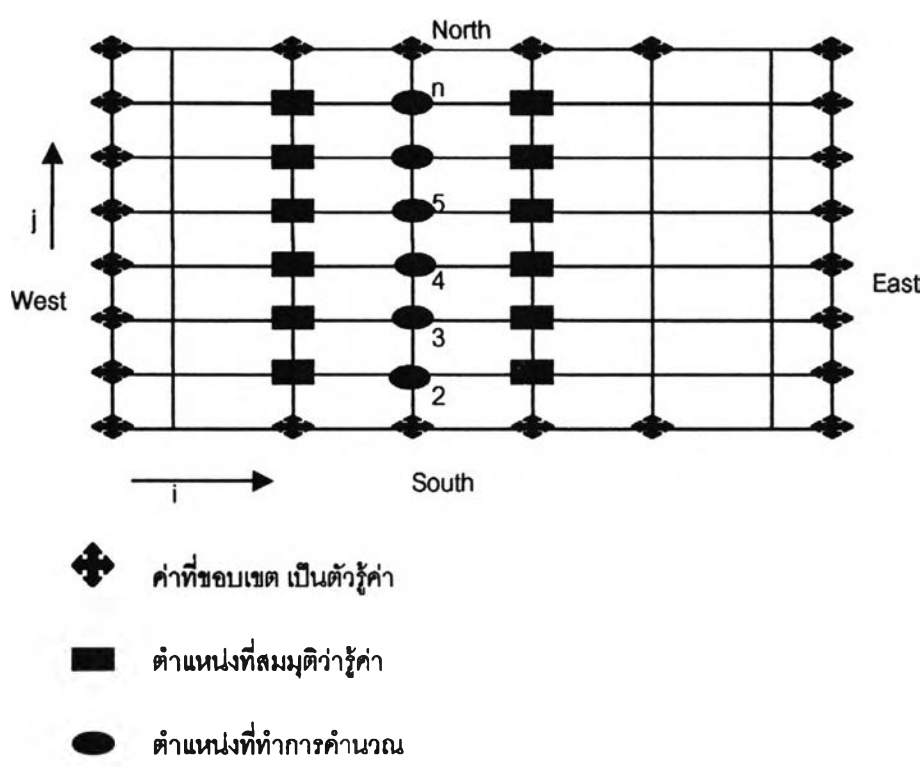
$$a_P\phi_P = a_E\phi_E + a_W\phi_W + a_N\phi_N + a_S\phi_S + b \quad \text{.....7}$$

ในการคำนวณหาคำตอบ อาจประยุกต์โดยการคำนวณทีละแถว เช่น จาก n ไป s

จัดรูปสมการที่ 7 ใหม่

$$-a_S\phi_S + a_P\phi_P - a_N\phi_N = a_E\phi_E + a_W\phi_W + b \quad \text{.....8}$$

โดยถือว่าเทอมทางขวามือเป็นเทอมที่รู้ค่า จัดรูปสมการที่ 8 ในอยู่ในรูปทั่วไปดังสมการที่ 2 เมื่อ $\alpha_j = a_N$, $\beta_j = a_S$, $D_j = a_P$ และ $C_j = a_W\phi_W + a_E\phi_E + b$ จากนั้นคำนวนหาคำตอบตั้งแต่ n ถึง s ค่าที่ j=2, 3, 4,... ,n ดังรูปที่ 1



รูปที่ 1 การคำนวนทีละแถวโดยใช้ TDMA

ในการจะคำนวณจากเหนือไปใต้ ของทิศตะวันตกไปทิศตะวันออก โดยในการคำนวณค่า ϕ_W ทางตะวันตกของตำแหน่ง P เป็นค่าที่รู้จากการคำนวณในแถวก่อนหน้าส่วนค่า ϕ_E เป็นค่าที่ยังคำนวณไปไม่ถึงดังนั้นต้องเดาค่า ϕ_E โดยใช้ค่า ϕ_E ที่ได้จากการคำนวณในรอบก่อนหน้าของการคำนวณ (ถ้าเป็นการคำนวณในรอบแรกมักเดาค่าให้เป็นศูนย์เสียก่อน) คำนวณเช่นนี้ไปเรื่อยๆ จนกระทั่งผลการคำนวณในแต่ละรอบไม่เปลี่ยนแปลง (convergence)

ภาคผนวก ค

เทคนิคการกระจายไฟไนต์ดิฟเฟอเรนซ์

ในการหาคำตอบสมการการถ่ายโอนความร้อนโดยวิธีเชิงวิเคราะห์ ถึงแม้จะได้คำตอบถูกต้องแม่นยำ แต่วิธีการคำนวณนั้นค่อนข้างยุ่งยาก ดังนั้นจึงใช้วิธีการคำนวณเชิงตัวเลขเข้ามาช่วย โดยใช้การแปลงสมการอนุพันธ์ย่อยให้อยู่ในรูปของไฟไนต์ดิฟเฟอเรนซ์ก่อน เพื่อนำไปหาคำตอบของสมการต่อไป

อนุกรมของ Taylor (Taylor Series)

อนุกรมของ Taylor เป็นพื้นฐานสำคัญอันหนึ่งของวิธีการทางตัวเลขที่ใช้ประยุกต์กับโจทย์ปัญหาการถ่ายเทความร้อน อนุกรมนี้เป็นการแสดงค่าฟังก์ชั่น y(x) ที่จุด $x+\Delta x$ ในรูปของ y และ derivatives ของ y ที่จุด x ดังนี้

$$y(x+\Delta x)=y(x)+\Delta x\frac{dy(x)}{dx}+\frac{(\Delta x)^2}{2!}\frac{d^2y(x)}{dx^2}+\ldots+\frac{(\Delta x)^n}{n!}\frac{d^ny(x)}{dx^n}\quad\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 1$$

หรือ นิยมเขียนให้กะทัดรัด โดยให้ $h=\Delta x$ และ $dy/dx=f'(x_i)$, $d^2y/dx^2=f''(x_i)$,...

$$f(x+h)=f(x)+hf'(x)+\frac{(h)^2}{2!}f''(x)+\ldots+\frac{(h)^n}{n!}f^n(x)\quad\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 2$$

ในการประมาณค่า เรามักใช้เพียง 2 เทอมแรกของอนุกรม Taylor เพื่อให้ง่ายต่อการคำนวณ เช่น

$$f(x+h)\cong f(x)+hf'(x)\quad\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 3$$

ในกรณีนี้เทอมที่ถูกตัดออก ประกอบด้วย

$$\frac{h^2}{2!}f''(x)+\frac{h^3}{3!}f'''(x)+\ldots.+\frac{h^n}{n!}f^n(x)$$

ก็คือความผิดพลาดจากการตัดเทอม ความผิดพลาดนี้ขึ้นอยู่กับขนาด h และจะน้อยลงถ้าค่า h เล็กลง และเขียนความผิดพลาดนี้ว่าแปรผันตาม ตัวเลขยกกำลังของ h ในเทอมแรกที่ถูกตัดออก เช่น $O(h^2)$ คือ ลำดับดับความผิดพลาดกำลัง 2 เนื่องจากค่า h มีค่าน้อยกว่าหนึ่ง ดังนั้น h ยกกำลังสูงๆ จะมีค่าน้อยกว่า h ยกกำลังต่ำๆ ก็คือไฟไนต์ดิฟเฟอเรนซ์ในลำดับสูงขึ้น จะให้ค่าความผิดพลาดน้อยลง

$$O(h^4)<O(h^2)<O(h)$$

ที่กล่าวมาเป็นการกระจายอนุกรมTaylor ไปข้างหน้าในทางเดียวกันสามารถกระจายย้อนกลับได้

$$f(x-h)=f(x)-hf'(x)+\frac{(h)^2}{2!}f''(x)+\ldots+\frac{(h)^n}{n!}f^n(x)\quad\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots 4$$

การกระจายแบบกึ่งกลางก็คือ เอาสมการที่ 2 ลบด้วยสมการที่ 4

$$f(x+h)-f(x-h)=2f'(x)+\frac{(h)^3}{3!}f'''(x)+O(h^2)\quad\ldots\ldots 5$$

สูตรไฟไนต์ดิฟเฟอเรนซ์ (Finite Difference Formulas)

1. **สูตรไฟไนต์ดิฟเฟอเรนซ์แบบไปข้างหน้า** (Forward Finite Difference Formulas)

ค่าความผิดพลาด

อนุพันธ์อันดับ 1

$$f'(x_i)\approx\frac{f(x_{i+1})-f(x_i)}{h}\quad\ldots\ldots O(h)$$

$$f'(x_i)\approx\frac{-f(x_{i+2})+4f(x_{i+1})-3f(x_i)}{2h}\quad\ldots\ldots O(h^2)$$

อนุพันธ์อันดับ 2

$$f''(x_i)\approx\frac{f(x_{i+2})-2f(x_{i+1})+f(x_i)}{h^2}\quad\ldots\ldots O(h)$$

$$f''(x_i)\approx\frac{-f(x_{i+3})+4f(x_{i+2})-5f(x_{i+1})+2f(x_i)}{h^2}\quad\ldots\ldots O(h^2)$$

2. **สูตรไฟไนต์ดิฟเฟอเรนซ์แบบย้อนหลัง** (Backward Finite Difference Formulas)

ค่าความผิดพลาด

อนุพันธ์อันดับ 1

$$f'(x_i)\approx\frac{f(x_i)-f(x_{i-1})}{h}\quad\ldots\ldots O(h)$$

$$f'(x_i)\approx\frac{3f(x_i)-4f(x_{i-1})+f(x_{i-2})}{2h}\quad\ldots\ldots O(h^2)$$

อนุพันธ์อันดับ 2

$$f''(x_i)\approx\frac{f(x_i)-2f(x_{i-1})+f(x_{i-2})}{h^2}\quad\ldots\ldots O(h)$$

$$f''(x_i)\approx\frac{2f(x_i)-4f(x_{i-1})+4f(x_{i-2})-2f(x_{i-3})}{h^2}\quad\ldots\ldots O(h^2)$$

3. สูตรไฟไนต์ดิฟเฟอเรนซ์แบบกึ่งกลาง (Centered Finite Difference Formulas)

ค่าความผิดพลาด

อุนพันธ์อันดับ 1

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_{i-1})}{2h} \quad \ldots\ldots O(h^2)$$

$$f'(x_i) \approx \frac{-f(x_{i+2}) + 8f(x_{i+1}) - 8f(x_{i-1}) + f(x_{i-2})}{12h} \quad \ldots\ldots O(h^4)$$

อนุพันธ์อันดับ 2

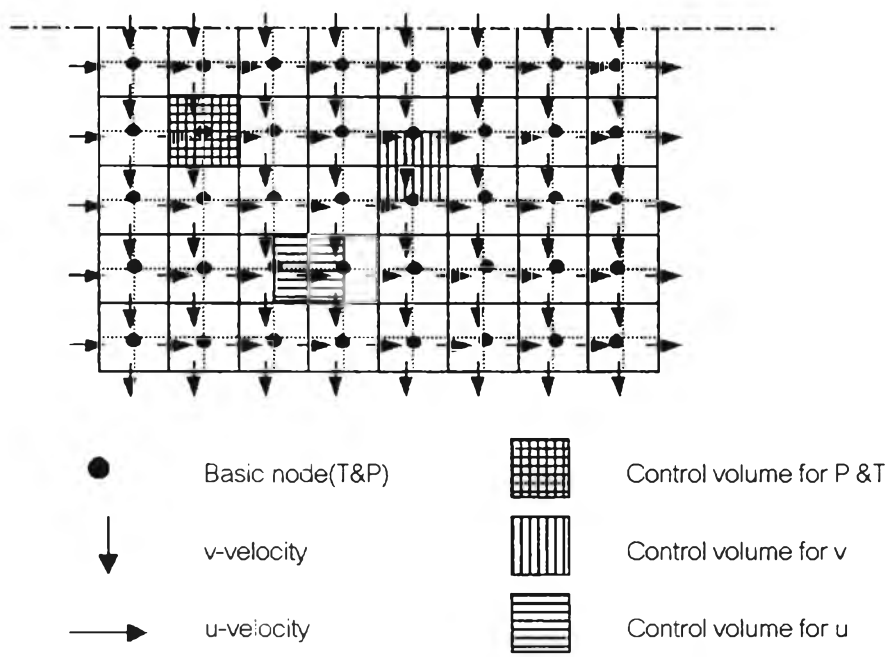
$$f''(x_i) \approx \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1})}{h^2} \quad \ldots\ldots O(h^2)$$

$$f''(x_i) \approx \frac{-f(x_{i+2}) + 16f(x_{i+1}) - 30f(x_i) + 16f(x_{i-1}) - f(x_{i-2})}{12h^2} \quad \ldots\ldots O(h^4)$$

ภาคผนวก ง

เทคนิค Staggered Grid

เทคนิค staggered grid ที่ใช้การกำหนดจุดของความเร็วกับความดันอยู่คนละตำแหน่ง โดยจะแสดงใน 2 มิติ คือในแนวแกนท่อ z และแนวรัศมี r (Versteeg และ Malalasekera, 1995)

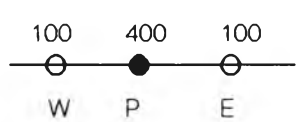


จากรูปเป็นการกำหนดปริมาตรควบคุม staggered โดยกำหนดส่วนปริมาตรของ z โมเมนตัมถูกแยกออกมาในทิศทางของแกน z และส่วนปริมาตรของ r โมเมนตัมก็ถูกแยกออกมาพิจารณาในทิศทาง r ผลทำให้ตำแหน่งของความเร็วไม่ตรงกับตำแหน่งของความดันและอุณหภูมิ วิธีนี้มีข้อดีคือ

1. ตัดปัญหาที่อาจเกิดขึ้นในกรณี คำตอบของความเร็วที่อาจเกิดการแกว่ง เช่น จากสมการ $du/dz=0$ ถ้าเราไม่คำนวณแบบ staggered ในการกระจายสมการความต่อเนื่องโดนใช้วิธีการกระจายแบบ centered difference มากระจายสมการได้

$$\frac{u_P + u_E}{2} - \frac{u_W + u_P}{2} = u_E - u_W = 0$$

ผลจากการกระจายสมการความต่อเนื่องทำให้ u_E มีค่าเท่ากับ u_W และอาจเกิดกรณีของความเร็ว เช่น



จากรูปคำตอบนี้ก็เป็นคำตอบหนึ่งของสมการความต่อเนื่อง ทั้งๆ ที่ความเป็นจริงค่าของ u_P ควรจะมีค่าเท่ากับ 100 และเป็นไปไม่ได้ที่ u_P จะเท่ากับ 400 นี่ก็เป็นข้อสังเกตหนึ่งที่อาจเกิดขึ้นได้ในการคำนวณปกติที่ไม่ใช่ staggered และเช่นเดียวกันในกรณีของความดันก็อาจเกิดเหตุการณ์เช่นเดียวกับความเร็ว ถ้าไม่ได้กำหนดตำแหน่งของกริดแบบ staggered ดังนั้นถ้าเรากำหนดสมการโมเมนตัมแบบ staggered ค่าความแตกต่างระหว่างความดัน 2 จุด ในกริด ก็คือแรงขับดัน (driving force) ที่ทำให้เกิดคอมโพเนนต์ความเร็วบนตำแหน่งระหว่างกริดของความเร็วนั่นเอง ดังนั้นการกำหนดแบบนี้จึงสมเหตุสมผลกว่า

2. การใช้วิธี staggered กริดก็คือสะดวกในการคำนวณสมการความต่อเนื่องและสมการอนุรักษ์พลังงานมากกว่าเพราะ ในสมการเหล่านี้ค่าความเร็วที่ใช้เป็นค่าที่อยู่ตรงรอยต่อของปริมาตรควบคุมของอุณหภูมิอยู่แล้วจึงสามารถนำมาใช้ได้เลย

ภาคผนวก จ

โปรแกรม

1. โปรแกรม Fortran

```
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                 Program for calculated Double-Pipe Heat Exchanger(DHEX)
C            @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C        1.Steady Stated used SIMPLE algorithm (Patankar ,1972)
C        2.Discretised convection term and conduction term by Power law scheme
C        3.Unstedy state used Implicit method
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

        PARAMETER (M=32,N=26)
        DIMENSION TOLD(N,M),TNEW(N,M),POLD(N,M),PNEW(N,M)
        DIMENSION ZOLD(N,M-1),ZNEW(N,M-1),ROLD(N+1,M),RNEW(N+1,M)
        REAL TOLD,TNEW,POLD,PNEW
        REAL ZOLD,ZNEW,ROLD,RNEW
        INTEGER N,M,ITNO,ITRESULT

        OPEN(UNIT=1,FILE='STEADYZ')
        OPEN(UNIT=2,FILE='STEADYR')
        OPEN(UNIT=3,FILE='STEADYP')
        OPEN(UNIT=4,FILE='STEADYT')
        OPEN(UNIT=5,FILE='STEADYERR')

        OPEN(UNIT=111,FILE='UNSTEDYZ')
        OPEN(UNIT=112,FILE='UNSTEDYR')
        OPEN(UNIT=113,FILE='UNSTEDYP')
        OPEN(UNIT=114,FILE='UNSTEDYT')
        OPEN(UNIT=115,FILE='UNSTEDYERR')
C       1.INPUT CASE OF STUDY
         CALL STUDYNO(STEAD)
c       2. INPUT CONDITION OF DHEX
        CALL INPCONST(QHIN,QCIN,THIN,TCIN,AL,DT)
        IF(STEAD.EQ.2.) THEN
        CALL NEWINPUT(QHIN,QCIN,THIN,TCIN,CHOICE)
        CALL ATTIME(TIMEEND)
        END IF
```

```
C       3.DHEX PARAMETER
CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
        ITNO    =1
        ITRESULT=1
        UNSTEADY=0.0
C       4.CONVERSE FLOW RATE TO VELOCITY
        CALL HOTVELO(QHIN,VHIN)
        CALL COLDVELO(QCIN,VCIN)

C       5.CALC.REYNOLD NO. AND HYDRODYNAMIC ENTRANCE LENGHT
C       CALL HREYNO(THIN,VHIN,HREY,HLEGHT)
C       CALL CREYNO(TCIN,VCIN,CREY,CLEGHT)

C       6.INITIAL GUESS CONDITION IN  INITIAL CONDITION
        CALL GUESSTNEW(THIN,TCIN,TNEW,N,M)
        CALL GUESSPNEW(PNEW,VHIN,THIN,VCIN,TCIN,AL,N,M)
        CALL GUESSRNEW(RNEW,N,M)
        CALL GUESSZNEW(ZNEW,VHIN,VCIN,AL,N,M)

        CALL GUESS(TNEW,TOLD,N,M,1,N,UNSTEADY)
        CALL GUESS(PNEW,POLD,N,M,1,N,UNSTEADY)
        CALL GUESS(RNEW,ROLD,N+1,M,1,N+1,UNSTEADY)
        CALL GUESS(ZNEW,ZOLD,N,M-1,1,N,UNSTEADY)
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                       7.CALC. STEADY STATE
        CALL EQUAT(UNSTEADY,ZNEW,ZOLD,RNEW,ROLD,PNEW,TNEW,TOLD,N,M,AL,DT)
        WRITE(111,*) 'ZFLOW IN STEADY '
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                       8.UNSTEADY STATE PARAMETER
        TIME    =0.0
999     UNSTEADY=1.0
        DT        =1.0
        TIME      =ITNO*DT
        TIMESTEP  =1.0
        ERRMIN    =0.00001
```

```
      ERROR    =0.0
C     9. GUESS NEW CONDITION OF UNSTEADY STATE
      CALL GUESS(TNEW,TOLD,N,M,1,N,UNSTEADY)
      CALL GUESS(PNEW,POLD,N,M,1,N,UNSTEADY)
      CALL GUESS(RNEW,ROLD,N+1,M,1,N+1,UNSTEADY)
      CALL GUESS(ZNEW,ZOLD,N,M-1,1,N,UNSTEADY)
      IF(CHOICE.EQ.1.) CALL THOTCHAGE(TNEW,THNEW,TCNEW,AL,N,M)
      IF(CHOICE.EQ.2.AND.TIME.LE.TIMESTEP) THEN
      CALL HOTNEW(ZNEW,N,M,ITNO,VHIN,QHNEW,TIMESTEP,TIME,DT)
      ENDIF

      IF(CHOICE.EQ.3.AND.TIME.LE.TIMESTEP) THEN
      CALL COLDNEW(ZNEW,N,M,ITNO,VCIN,QCNEW,TIMESTEP,TIME,AL,DT)
      ENDIF
      WRITE(*,*) 'ITNO =',ITNO,'          TIME =',TIME
C     10. CALC. UNSTEADY STATE
      CALL EQUAT(UNSTEADY,ZNEW,ZOLD,RNEW,ROLD,PNEW,TNEW,TOLD,N,M,AL,DT)
      CALL ERRCHECK(ZNEW,ZOLD,TNEW,TOLD,N,M,ERROR)
      WRITE(*,*) 'ERROR',ERROR
      WRITE(*,*) '--------------------'
9090  IF(ERROR.LE.ERRMIN) GOTO 10000
      CALL WTIME(ITNO,TIME)
      CALL WUNSTED(ZNEW,RNEW,PNEW,TNEW,N,M)
      ITNO    =ITNO+1
      ITRESULT=ITRESULT+1
      GOTO 999
      GOTO 9090
10000 CALL WENDPRG(TIME,ITNO)
      CALL WRESULT(ZNEW,RNEW,PNEW,TNEW,N,M)
      CALL WUNSTED(ZNEW,RNEW,PNEW,TNEW,N,M)
      WRITE(*,*)'TIME',TIME
      END
```

```
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
        SUBROUTINE ERRCHECK(ZNEW,ZOLD,TNEW,TOLD,N,M,ERROR)
        REAL ZNEW(N,M-1),ZOLD(N,M-1),TNEW(N,M),TOLD(N,M)
        CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
        IHIN    =1
        IHOUT   =NHIN-1
        ICIN    =NHOUT+1
        ICOUT   =NCIN-1
        JIN     =2
        JOUT    =M-1
        CALL ERRCALC(ZNEW,ZOLD,IHIN,IHOUT,JIN,JOUT,N,JOUT,ZERRHOT)
        CALL ERRCALC(ZNEW,ZOLD,ICIN,ICOUT,JIN,JOUT,N,JOUT,ZERRCLD)
        CALL ERRCALC(TNEW,TOLD,IHIN,N,JIN,JOUT,N,M,TERROR)
        WRITE(*,*) 'ZERRHOT ',ZERRHOT
        WRITE(*,*) 'ZERRCLD ',ZERRCLD
        WRITE(*,*) 'TERROR  ',TERROR
        IF(ZERRHOT.GT.ZERRCLD) THEN
        ERROR=ZERRHOT
        ELSE
        ERROR=ZERRCLD
        ENDIF
        IF(TERROR.GT.ERROR) ERROR=TERROR
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
        SUBROUTINE ERRCALC(A,B,IN,IOUT,JIN,JOUT,NROW,MCOL,ERROR)
        REAL A(NROW,MCOL),B(NROW,MCOL)
        REAL XDIMENS,YDIMENS,ALLCELL,AMEAN,ERROR
C   MEAN:
C   A(:)  :NEWMATRIX                                            ;INPUT
C   B(:)  :OLDMATRIX OR GUESSMATRIX                             ;INPUT
C   IN/JIN     :NO. START ROW / NO.START COLUMN TO CALC.        ;INPUT
C   IOUT/JOUT  :NO. END  ROW / NO. END COLUMN TO CALC.          ;INPUT
C   NROW       :NO.ROW OF A(:),B(:)                             ;INPUT
C   MCOL       :NO.COLUMN OF A(:), B(:)                         ;INPUT
```

```
C    ERROR        :CALCULATED PERCENT ERROR                          ;OUTPUT

                    ASUM=0.0
                    AERR=0.0
     DO 1 I=IN,IOUT
     DO 1 J=JIN,JOUT
     ADIFF   =ABS(A(I,J)-B(I,J))
     ASUM    =ASUM+ABS(A(I,J))
     AERR    =ADIFF+AERR
1    CONTINUE
                    XDIMENS=IOUT-IN+1.
                    YDIMENS=JOUT-JIN+1.
                    ALLCELL=XDIMENS*YDIMENS
                    AMEAN  =ASUM/ALLCELL
     IF(AMEAN.NE.0.) THEN
          ERROR=AERR/AMEAN/ALLCELL
     ELSE
          ERROR=0.0
     ENDIF
     RETURN
     END
C    @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C    FOR: WRITE TIME AND ITERATION NO. OF UNSTEADY STATE CASE
     SUBROUTINE WENDPRG(TIME,ITNO)
     REAL TIME
     WRITE(111,*) 'UNSTEADY STATE CONVERSE AT TIME=',TIME
     WRITE(112,*) 'UNSTEADY STATE CONVERSE AT TIME=',TIME
     WRITE(113,*) 'UNSTEADY STATE CONVERSE AT TIME=',TIME
     WRITE(114,*) 'UNSTEADY STATE CONVERSE AT TIME=',TIME
     WRITE(111,*) 'NO.ITERATION=',ITNO
     WRITE(112,*) 'NO.ITERATION=',ITNO
     WRITE(113,*) 'NO.ITERATION=',ITNO
     WRITE(114,*) 'NO.ITERATION=',ITNO
     RETURN
     END
```

```
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C       FOR: WRITE TIME AND ITERATION NO. OF STEADY STATE CASE
        SUBROUTINE WTIME(ITNO,TIME)
        REAL TIME
        WRITE (1,*) 'ITNO =',ITNO,'          TIME =',TIME
        WRITE (2,*) 'ITNO =',ITNO,'          TIME =',TIME
        WRITE (3,*) 'ITNO =',ITNO,'          TIME =',TIME
        WRITE (4,*) 'ITNO =',ITNO,'          TIME =',TIME
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
        SUBROUTINE HOTNEW(ZNEW,N,M,ITNO,VHIN,QHNEW,TIMESTEP,TIME,DT)
        REAL ZNEW(N,M-1)
        CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
          IF(ITNO.EQ.1) THEN
          CALL HOTVELO(QHNEW,VHNEW)
          HOTDIFF=(VHNEW-VHIN)/(TIMESTEP/DT)
          ENDIF
          IF(TIME.LE.TIMESTEP) THEN
          DO 1 I=1,NHIN-1
          ZNEW(I,1)       =ZNEW(I,1)+HOTDIFF
1         CONTINUE
          ENDIF
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
        SUBROUTINE COLDNEW(ZNEW,N,M,ITNO,VCIN,QCNEW,TIMESTEP,TIME,AL,DT)
        REAL ZNEW(N,M-1)
        CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
        IF(ITNO.EQ.1) THEN
          CALL COLDVELO(QCNEW,VCNEW)
          COLDIFF       =(VCNEW-VCIN)/(TIMESTEP/DT)
          VCLDNEW       =VCIN
        ENDIF
        IF(TIME.LE.TIMESTEP) THEN
```

```
      VCLDNEW     =VCLDNEW+COLDIFF
      DO 11 I=NHOUT+1,NCIN-1
          ZNEW(I,1)      =AL*VCLDNEW+(1.-AL)*ZNEW(I,1)
          ZNEW(I,M-1) =AL*ZNEW(I,M-1)+(AL-1.)*VCLDNEW
11    CONTINUE
     ENDIF
     RETURN
     END
C    @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C    FOR : CALC. VELOCITY, PRESSURE AND TEMPERATURE OF STEADY STATE AND
C        UNSTEADY STATE
     SUBROUTINE EQUAT(USTD,ZNEW,ZOLD,RNEW,ROLD,P,TNEW,TOLD,N,M,XAL,DT)
     REAL ZNEW(N,M-1),ZOLD(N,M-1)
     REAL RNEW(N+1,M),ROLD(N+1,M)
     REAL TNEW(N,M),TOLD(N,M)
     REAL P(N,M),PC(N,M)

     ITNO    =1
     ITRESULT=1
C    ERRMIN =0.00001
     ERRMIN =0.00000001
     TERR    =0.0
     ZERR    =0.0
     CONVH =0.0
     CONVC =0.0
     CONVERSEH=0.0
     CONVERSEC=0.0

     CALL GUESSZERO(PC,N,M,1,N)
     CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
C    ------------------------------------------
C         SIMPLE: HOT FLOW EQUATION
10000 IF(CONVH.NE.999) THEN
     AL           =1.0
     IST          =1
```

```
      IEND          =NHIN-1
      AMASSMIN      =0.000005
      WTZ           =0.8
      CALL FLDFLW(TNEW,TOLD,ZNEW,ZOLD,RNEW,ROLD,P,PC,N,M,AL,DT,IST,
     &              IEND,USTD,CONVH,ERRZH,AMASSMIN,SUMH,WTZ)
      ENDIF
C     ------------------------------------
C           SIMPLE: COLD FLOW EQUATION
      IF(CONVC.NE.999) THEN
            ALC           =XAL
            ISTC          =NHOUT+1
            IENDC         =NCIN-1
            AMASSMIN      =0.00003
            WTZC          =0.5
      CALL FLDFLW(TNEW,TOLD,ZNEW,ZOLD,RNEW,ROLD,P,PC,N,M,ALC,DT,ISTC,
     &              IENDC,USTD,CONVC,ERRZC,AMASSMIN,SUMC,WTZC)
      ENDIF
C     ------------------------------------
C           SIMPLE: HOT & COLD FLOW EQUATION
      IF(CONVERSEH.EQ.999.AND.CONVERSEC.EQ.999.) THEN
            AL            =1.0
            IST           =1
            IEND          =NHIN-1
            AMASSMIN      =0.000005
            WTZ           =0.8
      CALL FLDFLW(TNEW,TOLD,ZNEW,ZOLD,RNEW,ROLD,P,PC,N,M,AL,DT,IST,
     &              IEND,USTD,CONVERSEH,ERRZH,AMASSMIN,SUMH,WTZ)

C           ENERGY EQUATION
C     INCASE OF TEST PRESSURE TERM
      ITST    =1
      ITEND   =N
      CALL ENERGY(USTD,P,ZNEW,RNEW,TOLD,TNEW,N,M,DT,XAL,ITST,ITEND,TERR)
C     ------------------------------------
C           CHECK ERROR
```

```
C       IF(ERRZH.GE.ERRZC) ZERR=ERRZH*100.
C       IF(ERRZH.LT.ERRZC) ZERR=ERRZC*100.
        IF(ERRZH.GE.ERRZC) ZERR=ERRZH
        IF(ERRZH.LT.ERRZC) ZERR=ERRZC
        IF(ZERR.GE.TERR) THEN
        ERROR=ZERR
        ELSE
        ERROR=TERR
        ENDIF
9000    IF(ERROR.LE.ERRMIN) GOTO 999
        ITNO          =ITNO+1
        GOTO 10000
        GOTO 9000
999     WRITE(*,*) 'ITNO ',ITNO,' CONVSE HOT ',SUMH,' CONVSE COLD',SUMC
        RETURN
END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C       FOR: CALC. ONLY VELOCITY AND PRESSURE (SIMPLE ALGORITHM)
SUBROUTINE FLDFLW(TNEW,TOLD,ZNEW,ZOLD,RNEW,ROLD,P,PC,N,M,AL,DT
   &    ,IST,IEND,USTDY,CONVSE,ZERR,AMASSMIN,SUM,WTZ)
        REAL TNEW(N,M),TOLD(N,M),USTDY
        REAL ZNEW(N,M-1),RNEW(N+1,M),P(N,M)
        REAL ZGUSS(N,M-1),PC(N,M)
        REAL ZOLD(N,M-1),ROLD(N+1,M)
        REAL APU(N,M-1),PRX(N+1,M)
C   MEAN:
C       PC(:) : CORRECTION PRESSURE MATRIX
C       APU(:): COEFT. OF PRESSURE DIVIDE COEFT ZNEW(:) SET IN MATRIX
C       PRX(:): COEFT. OF PRESSURE DIVIDE COEFT RNEW(:) SET IN MATRIX
C       USTDY : UNSTEADY STATE OR STEADY STATE(0=UNSTEADY,1=STEADY)
C       CONVSE: CONVERGENCE IF CONTINUTY CONVERSE TO ZERO YET
C       IST   :  START ROW TO CALCULATE
C       IEND  : END   ROW TO CALCULATE

C       UNSTEADY EQUAL 1.0 MEAN MATRIXGUESS(:) = MATRIXNEW(:)
```

```
      UNSTEADY=1.0
      DO 1 I=IST,IEND
      DO 1 J=1,M
      PC(I,J)=0.75*PC(I,J)
1     CONTINUE
      CALL GUESS(ZNEW,ZGUSS,N,M-1,IST,IEND,UNSTEADY)
      CALL ZMOMT(ZNEW,RNEW,ZOLD,TNEW,TOLD,P,APU,N,M,AL,DT,IST,IEND,WTZ)

      CALL RMOMT(RNEW,ZGUSS,ROLD,TNEW,TOLD,P,PRX,N,M,AL,DT,IST+1,IEND)
      CALL PCORT(ZNEW,RNEW,TNEW,TOLD,PC,APU,PRX,N,M,AL,DT
     &                ,IST,IEND,USTDY,CONVSE,AMASSMIN,SUM)
      CALL CORECT(ZNEW,RNEW,P,PC,APU,PRX,AL,IST,IEND,N,M)
      CALL ERRCALC(ZNEW,ZGUSS,IST,IEND,2,M-2,N,M-1,ZERR)
      IF(CONVSE.EQ.999) ZERR=0.0
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                          FOR CALC. ENERGY
C          @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE ENERGY(USTD,P,ZNEW,RNEW,TOLD,TNEW,N,M,DT,XAL
      &     ,IST,IEND,TERR)
      REAL ZNEW(N,M-1),RNEW(N+1,M),TOLD(N,M),TNEW(N,M),TGUESS(N,M)
      REAL P(N,M)
      REAL PW(M),PE(M),PP(M),PB(M),A(M),C(M)

                  ITNO    =0
                  FACT    =0.6
                  ERRMIN =0.00001
                  WT           =0.8
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      CALL GUESS(TNEW,TGUESS,N,M,1,N,1.)
C     ---------------------------------------
50    SUMDIFF =0.0
          TOTAL   =0.0
C     Assume TDMA COEFFICIENT MATRIXS ARE ZERO.
```

```
            CALL ZERO(PW,PE,PP,PB,A,C,M)
      DO 1 I=IST,IEND
      DO 2 J=2,M-1
                  TPO     =TOLD(I,J)
            IF(I.LE.NHOUT)  AL=1.0
                  IF(I.GT.NHOUT) AL=XAL
C     B.C
      TNEW(I,M)       =AL*TNEW(I,M-1)+(1.-AL)*TNEW(I,M)
      TNEW(I,1)       =AL*TNEW(I,1)  +(1.-AL)*TNEW(I,2)
C                     ----------
      CALL RADIAL(I,RN,RS,R)
      CALL ZFLOW(ZNEW,N,M,I,J,UP,UE,UW,UN,US)
      CALL FLOW(RNEW,N+1,M,I,J,VP,VE,VW,VN,VS)
      CALL ATEMP(TGUESS,N,M,I,J,TP,TE,TW,TN,TS)
      CALL TDIFFUS(TGUESS,ZNEW,N,M,I,J,R,RN,RS,DE,DW,DN,DS)
      CALL TCONVEC(UP,UW,VP,VN,TP,TE,TW,TN,TS,R,RN,RS,AE,AW,AN,AS)
      CALL TSOURCE(ZNEW,RNEW,TP,N,M,I,J,SOUCE)
C           ----------
                  FW=AW
                  FE=AE
C           ----------
      CALL TTIME(I,TP,TPO,DT,R,COEFTP,COEFTPO)
      CALL TCOEFT(I,J,M,AE,AW,AN,AS,DE,DW,DN,DS,COEFTP,COEFTPO
     &     ,SOUCE,TPO,AL,AP,AB)
      AMASS=(RN*VN-RS*VP)*DZ+R*(UP-UW)
      AB       =AB-P(I,J)*AMASS
      CALL ATEMP(TNEW,N,M,I,J,TP,TE,TW,TN,TS)

            CALL TDMA1(AW,AE,AP,WT,PW(J),PE(J),PP(J),M,J)
            CONST =AP*(1.-WT)/WT*TGUESS(I,J)
            PB(J)    =AB+AN*TN-AS*TS+CONST
            PB(2)    =PB(J)+TW*AL*(2.*DW+FW)
            PB(M-1) =PB(J)+TE*(2.*DE-FE)*(1.-AL)
            CALL TDMA3(PW(J),PE(J),PP(J),PB(J),A(J-1),C(J-1),A(J),C(J))
2        CONTINUE
```

```
      DO 10 L=M-1,2,-1
      ANS          =A(L)*TNEW(I,L+1)+C(L)
      DIFF         =ABS(TNEW(I,L)-ANS)
      SUMDIFF      =SUMDIFF+DIFF
      TNEW(I,L)    =FACT*ANS+(1.-FACT)*TNEW(I,L)
      TOTAL        =TOTAL+TNEW(I,L)
10    CONTINUE
1     CONTINUE
          AGRID        =(IEND-IST+1.)*(M-2.)
          TMEAN        =TOTAL/AGRID
          ERROR        =SUMDIFF/TMEAN/AGRID
      ITNO=ITNO+1
40  IF(ERROR.LE.ERRMIN) GOTO 60
      GOTO 50
      GOTO 40
C60   WRITE(*,*) 'TEMPERA',ITNO,ERROR
60    CALL ERRCALC(TNEW,TGUESS,IST,IEND,2,M-1,N,M,TERR)
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR : CALC. SOURCE TERM OF ENERGY EQUATION
      SUBROUTINE    TSOURCE(ZNEW,RNEW,TP,N,M,I,J,SOURCE)
      REAL ZNEW(N,M-1),RNEW(N+1,M)
      CALL RRADIA(I,RP,RRS,RR)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      CALL ZFLOW(ZNEW,N,M,I,J,UP,UE,UW,UN,US)
      CALL FLOW(RNEW,N+1,M,I,J,VP,VE,VW,VN,VS)
C         ------------------------
              VS=VP
              UE=UP
          UZN=0.25*(UE+UW+UN+ZNEW(I+1,J-1))
          UZS=0.25*(UE+UW+US+ZNEW(I-1,J-1))
          VRE=0.25*(VN+VS+VE+RNEW(I+1,J+1))
          VRW=0.25*(VN+VS+VW+RNEW(I+1,J-1))
```

```
C            ____________________
             IF(I.EQ.1) UZS=0.5*(UE+UW)
             VIS=FVIS(TP)
      AVRSQ          =2.*VIS*DR*DZ/RP*(0.5*(VN+VS))**2.
      DUDRDVDZ       =2.*VIS*RP*(UZN-UZS)*(VRE-VRW)
      DVDRSQ         =2.*VIS*RP*DZ/DR*(VN-VS)*(VN-VS)
      DUDZSQ         =2.*VIS*RP*DR/DZ*(UE-UW)*(UE-UW)
      DUDRSQ         =  VIS*RP*DZ/DR*(UZN-UZS)*(UZN-UZS)
      DVDZSQ         =  VIS*RP*DR/DZ*(VRE-VRW)*(VRE-VRW)
      SOURCE         =DVDRSQ+AVRSQ+DUDZSQ+DUDRSQ+DUDRDVDZ+DVDZSQ
      IF(I.GE.NHIN.AND.I.LE.NHOUT)   SOURCE=0.0
      IF(I.GE.NCIN.AND.I.LE.NCOUT)   SOURCE=0.0
      RETURN
      END
c     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
c     FOR: USED POWER LAW CALC. COEFICIENT OF TEMPERATURE
      SUBROUTINE TCOEFT(I,J,M,FE,FW,FN,FS,DE,DW,DN,DS,COEFTP,
   &   COEFTPO,SOURCE,AOLD,AL,AP,AB)

      CALL POWLAW(FE,FW,FN,FS,DE,DW,DN,DS,ELN,ELS)


      AE      =-FE*(1.-AL)+DE
      AW      = FW*AL+DW
      AN      =-FN*(1.-ELN)+DN
      AS      = FS*ELS+DS

      IF(J.EQ.2)        AW=0.0
      IF(J.EQ.(M-1))    AE=0.0
      IF(I.EQ.1)        AS=0.0

      AP      = AE+AW+AN+AS+(FE-FW)+(FN-FS)+COEFTP
      AB      = COEFTPO*AOLD+SOURCE
      IF(J.EQ.2)        AP=AP+AL*(FW+2.*DW)
      IF(J.EQ.(M-1))    AP=AP+(1.-AL)*(2.*DE-FE)
```

```
      FE    =AE
      FW    =AW
      FN    =AN
      FS    =AS
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR: CALC.  COEFICIENT OF TIME TERM.
      SUBROUTINE    TTIME(I,TP,TPO,DT,R,COEFTP,COEFTPO)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)

      IF(I.LT.NHIN) THEN
            DENCP   =FD(TP)*FCP(TP)
            DENCPO =FD(TPO)*FCP(TPO)
            ELSE
            IF(I.GT.NHOUT.AND.I.LT.NCIN) THEN
            DENCP  =FD(TP)*FCP(TP)
            DENCPO        =FD(TPO)*FCP(TPO)
            ELSE
            DENCP  =FDS(TP)*FCPS(TP)
            DENCPO        =FDS(TPO)*FCPS(TPO)
            ENDIF
      ENDIF
      IF(DT.NE.0.0) THEN
            RDZBYDT       =R*DZ/DT
            COEFTP=DENCP *RDZBYDT
            COEFTPO       =DENCPO*RDZBYDT
      ELSE
            COEFTP=0.0
            COEFTPO       =0.0
      ENDIF
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR : CALC. CONVECTION TERM OF ENERGY EQ.
```

```
      SUBROUTINE TCONVEC(UP,UW,VP,VN,TP,TE,TW,TN,TS,R,RN,RS,FE,FW,FN,FS)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)

      APP     =FD(TP)*FCP(TP)
      FE      =UP*R*0.5*(APP+FD(TE)*FCP(TE))
      FW      =UW*R*0.5*(APP+FD(TW)*FCP(TW))
      FN      =VN*RN*DZ*0.5*(APP+FD(TN)*FCP(TN))
      FS      =VP*RS*DZ*0.5*(APP+FD(TS)*FCP(TS))
      RETURN
      END
C
C     FOR: CALC. DIFFUSION TERM OF ENERGY EQ.
      SUBROUTINE TDIFFUS(TNEW,ZNEW,N,M,I,J,R,RN,RS,DE,DW,DN,DS)
      REAL TNEW(N,M),ZNEW(N,M-1)

      CALL ATEMP(TNEW,N,M,I,J,TP,TE,TW,TN,TS)
      CALL CONDUC(TP,TE,TW,TN,TS,I,AKE,AKW,AKN,AKS)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)

      RBYDZ           =R/DZ
      DZBYDR          =DZ/DR
      DE=AKE*RBYDZ
      DW=AKW*RBYDZ
      DN=AKN*RN*DZBYDR
      DS=AKS*RS*DZBYDR
      IF(I.EQ.(NHIN-1).OR.I.EQ.NHOUT.OR.I.EQ.(NCIN-1)) THEN
      CALL HCOEFT(ZNEW,TNEW,N,M,I,J,H)
      DN      =2.*H*RN*DZBYDR
C     DN      =0.0
      ENDIF
      IF(I.EQ.NHIN.OR.I.EQ.(NHOUT+1).OR.I.EQ.NCIN) THEN
      CALL HCOEFT(ZNEW,TNEW,N,M,I,J,H)
      DS      =2.*H*RS*DZBYDR
C     DS      =0.0
      ENDIF
```

```
      IF(I.EQ.1)         DS=0.0
      IF(I.EQ.NCOUT)  DN=0.0
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR: FIND THERMAL CONDUCTIVITY AT INTERFACE OF FLUID AND WALL
      SUBROUTINE HCOEFT(ZNEW,TNEW,N,M,I,J,H)
      REAL ZNEW(N,M-1),TNEW(N,M)

      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      IF(I.LE.NHIN) THEN
      IX      =1
      IY      =NHIN-1
      TF      =TNEW(NHIN-1,J)
      CALL TBULK(TNEW,ZNEW,N,M,IX,IY,TB)
      Pr      =FVIS(TB)*FCP(TB)/FK(TB)
      H       =FVIS(TF)*FCP(TF)/Pr
      ELSE
      IX      =NHOUT+1
      IY      =NCIN-1
      CALL TBULK(TNEW,ZNEW,N,M,IX,IY,TB)
      Pr  =FVIS(TB)*FCP(TB)/FK(TB)
      IF(I.EQ.NHOUT.OR.I.EQ.(NHOUT+1))       TF=TNEW(NHOUT+1,J)
      IF(I.EQ.NCIN.OR.I.EQ.(NCIN-1))         TF=TNEW(NCIN-1,J)
      H       =FVIS(TF)*FCP(TF)/Pr
      ENDIF
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR : CALC. BULK TEMPERATURE FOR FIND THERMALCONDUCTIVITY AT INTERFACE
C           FLUID AND WALL
      SUBROUTINE TBULK(TNEW,ZNEW,N,M,IX,IY,TB)
      REAL TNEW(N,M),ZNEW(N,M-1)

                  XIN            =0.0
```

```
                    XOUT          =0.0
                    TSUMIN        =0.0
                    TSUMOUT       =0.0

        DO 1 I=IX,IY
        CALL RADIAL(I,RN,RS,R)
        TFIN    =TNEW(I,1)
        TFOUT   =TNEW(I,M)
        AIN     =2.*3.14*R*ZNEW(I,1)* FD(TFIN)*FCP(TFIN)
        AOUT    =2.*3.14*R*ZNEW(I,M-1)*FD(TFOUT)*FCP(TFOUT)
        XIN     =AIN    +XIN
        XOUT    =AOUT   +XOUT
        TSUMIN          =AIN*TFIN  +TSUMIN
        TSUMOUT         =AOUT*TFOUT+TSUMOUT
1       CONTINUE
        TBIN    =TSUMIN/XIN
        TBOUT   =TSUMOUT/XOUT
        TB      =(TBIN+TBOUT)/2.
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C       FOR : THERMAL CONDUCTIVITY OF FLUID AND SOLID
        SUBROUTINE CONDUC(TP,TE,TW,TN,TS,I,AKE,AKW,AKN,AKS)
        CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
        IF(I.LT.NHIN) THEN
                CALL FLUIDK(TP,TE,TW,TN,TS,AKE,AKW,AKN,AKS)
                ELSE
                IF(I.GT.NHOUT.AND.I.LT.NCIN) THEN
        CALL FLUIDK(TP,TE,TW,TN,TS,AKE,AKW,AKN,AKS)
        ELSE
                CALL SOLIDK(TP,TE,TW,TN,TS,AKE,AKW,AKN,AKS)
                ENDIF
         ENDIF
        RETURN
        END
```

```
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C       FOR: FIND THERMAL CONDUCTIVITY OF FLUID
        SUBROUTINE FLUIDK(TP,TE,TW,TN,TS,AKE,AKW,AKN,AKS)
        FKTP=FK(TP)
        AKE     =0.5*(FKTP+FK(TE))
        AKW     =0.5*(FKTP+FK(TW))
        AKN     =0.5*(FKTP+FK(TN))
        AKS     =0.5*(FKTP+FK(TS))
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C       FOR: FIND THERMAL CONDUCTIVITY OF SOLID
        SUBROUTINE SOLIDK(TP,TE,TW,TN,TS,AKE,AKW,AKN,AKS)
        FKSTP=FKS(TP)
        AKE  =0.5*(FKSTP+FKS(TE))
        AKW  =0.5*(FKSTP+FKS(TW))
        AKN  =0.5*(FKSTP+FKS(TN))
        AKS  =0.5*(FKSTP+FKS(TS))
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C       FOR: SOLVE PRESSURE CORRECTION EQUATION (DERIVED FROM CONTINUETY)
        SUBROUTINE PCORT(ZNEW,RNEW,TNEW,TOLD,PC,APU,PRX,N,M,AL,DT
   &                        ,IST,IEND,UNSTEADY,CONVERSE,AMASSMIN,SUM)
        REAL ZNEW(N,M-1),RNEW(N+1,M),TNEW(N,M),TOLD(N,M),PC(N,M)
        REAL APU(N,M-1),PRX(N+1,M)
        REAL PW(M),PE(M),PP(M),PB(M),A(M),C(M),AGRID,ERROR

                              ITNO          =0
                              FACT          =0.8
                              ERRMIN        =0.005
C       WARNING: NOT CHANGE         WT VALUE
                              WT                    =1.0
                IF(UNSTEADY.EQ.0.0)THEN
                CALL PRATIO(ZNEW,TNEW,IST,IEND,N,M,AL,RATIO)
```

```
                ELSE
                RATIO           =1.0
                ENDIF
C       ------------------------------------------
50      SUMDIFF         =0.0
        TOTAL   =0.0
        SUM             =0.0
        CALL ZERO(PW,PE,PP,PB,A,C,M)
        DO 1 I=IST,IEND
        DO 2 J=2,M-1
        CALL ATEMP(TNEW,N,M,I,J,TP,TE,TW,TN,TS)
        APUP    =APU(I,J)
        APUW    =APU(I,J-1)
        PRXN    =PRX(I+1,J)
        PRXP    =PRX(I,J)
        DENOLD          =FD(TOLD(I,J))
        DEN             =FD(TP)
        DENN    =FD(TN)
        DENS    =FD(TS)
        DENE    =FD(TE)
        DENW    =FD(TW)

        CALL RADIAL(I,RN,RS,R)
        CALL ZFLOW(ZNEW,N,M,I,J,UP,UE,UW,UN,US)
        CALL FLOW(RNEW,N+1,M,I,J,VP,VE,VW,VN,VS)
        CALL AEWP(DEN,DENE,DENW,APUP,APUW,J,M,R,AE,AW)
        CALL ANSP(DEN,DENN,DENS,PRXN,PRXP,I,IST,IEND,RN,RS,AN,AS)
        CALL ABP(DEN,DENE,DENW,DENN,DENS,DENOLD,UW,UP,VN,VP,
     &          J,M,DT,RN,RS,R,AL,AB,RATIO)
        CALL APP(AE,AW,AN,AS,AP)
                PB(J)  =AB+AN*PC(I+1,J)+AS*PC(I-1,J)
        CALL TDMA1(AW,AE,AP,WT,PW(J),PE(J),PP(J),M,J)
        CALL TDMA3(PW(J),PE(J),PP(J),PB(J),A(J-1),C(J-1),A(J),C(J))
        SUM =SUM+ABS(AB)
2       CONTINUE
```

```
      DO 10 L=M-1,2,-1
      ANS          =A(L)*PC(I,L+1)+C(L)
      DIFF         =ABS(PC(I,L)-ANS)
      SUMDIFF      =SUMDIFF+DIFF
      PC(I,L)      =FACT*ANS+(1.-FACT)*PC(I,L)
      TOTAL        =TOTAL+ABS(PC(I,L))
10    CONTINUE
1     CONTINUE
          AGRID          =(IEND-IST+1.)*(M-2.)
          PMEAN          =TOTAL/AGRID
          IF(PMEAN.NE.0.0)       ERROR        =SUMDIFF/PMEAN/AGRID
          IF(PMEAN.EQ.0.0)       ERROR        =SUMDIFF/AGRID
      ITNO=ITNO+1
40    IF(ERROR.LE.ERRMIN) GOTO 60
      GOTO 50
      GOTO 40
C60   WRITE(*,*) 'PCORECT',ITNO,ERROR
C     WRITE(*,*) 'CONVRSE',ITNO,SUM
60    IF(SUM.LT.AMASSMIN) THEN
      WRITE(5,*) 'FLOW IS CONVERSE AT'
      WRITE(5,*) SUM
          DO 99 IC=IST,IEND
          DO 99 JC=1,M
              PC(IC,JC)=0.0
99        CONTINUE
          CONVERSE=999.
      ELSE
          CONVERSE=0.0
      ENDIF
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR: FIND COEFFICIENT AT EAST AND WEST  OF PRESSURE CORRECTION TERM
      SUBROUTINE AEWP(DEN,DENE,DENW,APUP,APUW,J,M,R,AE,AW)
```

```
      DE=    0.5*(DEN+DENE)
      DW=    0.5*(DEN+DENW)
      AE=    R*DE*APUP
      AW=    R*DW*APUW
      IF(J.EQ.2)        AW=0.0
      IF(J.EQ.(M-1))    AE=0.0
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR: FIND COEFFICIENT AT NORTH AND SOUTH  OF PRESSURE CORRECTION TERM
      SUBROUTINE ANSP(DEN,DENN,DENS,PRXN,PRXP,I,IST,IEND,RN,RS,AN,AS)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      DN     =0.5*(DEN+DENN)
      DS     =0.5*(DEN+DENS)
      AN     =RN*DN*DZ*PRXN
      AS     =RS*DS*DZ*PRXP
      IF(I.EQ.IST)      AS=0.0
      IF(I.EQ.IEND)     AN=0.0
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR: FIND  HELP TERM TO CONVERGENCE :MASS CONSERVE (MASS IN / MASS OUT)
      SUBROUTINE PRATIO(ZNEW,TNEW,IST,IEND,N,M,AL,RATIO)
      REAL ZNEW(N,M-1),TNEW(N,M)
            SUMIN        =0.0
            SUMOUT       =0.0
      DO 1 I=IST,IEND
      VZIN    =ZNEW(I,1)
      VZOUT   =ZNEW(I,M-1)
      TZIN    =TNEW(I,1)
      TZOUT   =TNEW(I,M-1)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      CALL RADIAL(I,RN,RS,R)
      XIN     =VZIN *2.*3.14*R*FD(TZIN)
      XOUT    =VZOUT*2.*3.14*R*FD(TZOUT)
```

```
      SUMIN   =XIN     +SUMIN
      SUMOUT           =XOUT  +SUMOUT
1     CONTINUE
      IF(AL.EQ.1) RATIO=SUMIN/SUMOUT
      IF(AL.NE.1)       RATIO=SUMOUT/SUMIN
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR: FIND SOURCE TERM OF PRESSURE CORRECTION ( REACH TO ZERO IF PROGRAM
C         CONVERSE)
      SUBROUTINE ABP(DP,DE,DW,DN,DS,DOLD,UW,UP,VN,VP,J,M,DT
     &                        ,RN,RS,R,AL,AB,RATIO)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      IF(J.EQ.2)          UW=AL*UW+(1.-AL)*UP*RATIO
      IF(J.EQ.(M-1))     UP=AL*UW*RATIO+(1.-AL)*UP
      BN        =-RN*0.5*(DN+DP)*DZ*VN
      BS        = RS*0.5*(DS+DP)*DZ*VP
      BE        =-R *0.5*(DE+DP)*UP
      BW        = R *0.5*(DW+DP)*UW
      IF(DT.EQ.0.0) CONST=0.0
      IF(DT.NE.0.0) CONST=(DOLD-DP)*R*DZ/DT
      AB        =BN+BS+BE+BW+CONST
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR : FIND COEFFICIENT OF P TERM
      SUBROUTINE APP(AE,AW,AN,AS,AP)
   AP =AE+AW+AN+AS
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR : USED TDMA TO SOLVE MATRIX
      SUBROUTINE TDMA1(AW,AE,AP,WT,PW,PE,PP,MCOL,J)
      PW        =AW
      PE        =AE
```

```
      PP      =APWT
      IF(J.EQ.2)              PW=0.0
      IF(J.EQ.(MCOL-1))       PE=0.0
      RETURN
      END
C     [illegible]
      SUBROUTINE TDMA2(CONST,AN,AS,AE,AW,AB,UN,US,UE,UW,PB,MCOL,J)
                  PB      =AB+AN*UN+AS*US+CONST
      IF(J.EQ.2)              PB      =PB+AW*UW
      IF(J.EQ.(MCOL-1))       PB      =PB+AE*UE
      RETURN
      END
C     [illegible]
      SUBROUTINE TDMA3(APW,APE,APP,APB,AOLD,COLD,AVAL,CVAL)
      AVAL=APE/(APP-APW*AOLD)
      CVAL=(APW*COLD+APB)/(APP-APW*AOLD)
      RETURN
      END
C     [illegible]
C     FOR : USED POWER LAW TO CALC. CONDUCT. AND CONVEC. TERM
      SUBROUTINE POWLAW(FE,FW,FN,FS,DE,DW,DN,DS,ELN,ELS)
      IF(FN.GE.0.0)  ELN=1.0
      IF(FN.LT.0.0)  ELN=0.0
      IF(FS.GE.0.0)  ELS=1.0
      IF(FS.LT.0.0)  ELS=0.0
      CALL SUBPOWLAW(FE,DE)
      CALL SUBPOWLAW(FW,DW)
      CALL SUBPOWLAW(FN,DN)
      CALL SUBPOWLAW(FS,DS)
      RETURN
      END
C     [illegible]
      SUBROUTINE SUBPOWLAW(CONVEC,DIFFUS)
      IF(DIFFUS.NE.0.0) THEN
            POWERLAW=CONVEC/DIFFUS
```

```
            IF(ABS(POWERLAW).LE.10.) THEN
            DIFFUS=DIFFUS*(1.-0.1*ABS(POWERLAW))**5.
            ELSE
            DIFFUS=0.0
            ENDIF
      ENDIF
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR: SOLVE R-MOMENTUM
      SUBROUTINE RMOMT(RNEW,ZNEW,ROLD,T,TOLD,P,PRX,N,M,AL,DT,IST,IEND)
      REAL RNEW(N+1,M),ROLD(N+1,M),RGUESS(N+1,M),PRX(N+1,M)
      REAL ZNEW(N,M-1),T(N,M),TOLD(N,M),P(N,M)
      REAL PW(M),PE(M),PP(M),PB(M),A(M),C(M)
      REAL AGRID,ANS
                        ITNO   =0
                        FACT   =0.8
                        ERRMIN =0.00001
                        WT             =0.5
      UNSTEADY=1.0
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      CALL GUESS(RNEW,RGUESS,N+1,M,1,N+1,UNSTEADY)
      CALL GUESSZERO(PRX,N+1,M,1,N+1)
C     ---------------------------------
50    SUMDIFF        =0.0
      TOTAL  =0.0
      CALL ZERO(PW,PE,PP,PB,A,C,M)
      DO 1 I=IST,IEND
      DO 2 J=2,M-1
                        AOLD   =ROLD(I,J)
                        USW            =ZNEW(I-1,J-1)
                        DENOLD         =0.5*(FD(TOLD(I,J))+FD(TOLD(I-1,J)))
                        TP             =T(I,J)
                        TS             =T(I-1,J)
      CALL RADIAL(I,RN,RRP,R)
```

```
        CALL RRADIA(I,RRN,RRS,RR)
        CALL ZFLOW(ZNEW,N,M,I,J,UP,UE,UW,UN,US)
        CALL FLOW(RGUESS,N+1,M,I,J,VP,VE,VW,VN,VS)
        CALL RDIFFUS(T,N,M,I,J,RRP,RRN,RRS,RR,RN,DE,DW,DN,DS,DNP,DSP)
        CALL RCONVEC(T,UP,UW,US,USW,VP,VN,VS,RRN,RRS,RR,N,M,I,J,
     &  AE,AW,AN,AS)
        CALL RTIME(TP,TS,DENOLD,RR,DT,SNEW,SOLD)
        CALL RSOURCE(UP,UW,US,USW,VP,VN,VS,DE,DW,DN,DS,DNP,DSP,SOUCE)
        CALL RCOEFT(J,M,AE,AW,AN,AS,DE,DW,DN,DS,DNP,DSP,SNEW,SOLD,SOUCE
     &  ,AOLD,AL,AP,AB)
        CALL FLOW(RNEW,N+1,M,I,J,VP,VE,VW,VN,VS)
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C       HELP PARAMETER IN CASE VR(:) EQUAL ZERO ( IN CASE ONE DIMENTION)
                                IF(AP.EQ.0.0) AP=1.00
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
             PRX(I,J) =RRP*DZ*WT/AP
             WTX      =AP*(1.-WT)/WT*RGUESS(I,J)
             CONST    =RRP*DZ*(P(I-1,J)-P(I,J))+WTX

        CALL TDMA1(AW,AE,AP,WT,PW(J),PE(J),PP(J),M,J)
        CALL TDMA2(CONST,AN,AS,AE,AW,AB,VN,VS,VE,VW,PB(J),M,J)
        CALL TDMA3(PW(J),PE(J),PP(J),PB(J),A(J-1),C(J-1),A(J),C(J))
2       CONTINUE
        DO 10 L=M-1,2,-1
        ANS          =A(L)*RNEW(I,L+1)+C(L)
        RNEW(I,L)    =FACT*ANS+(1.-FACT)*RNEW(I,L)
        DIFF         =ABS(RNEW(I,L)-ANS)
        SUMDIFF      =SUMDIFF+DIFF
        TOTAL        =TOTAL+ABS(RNEW(I,L))
10      CONTINUE
             RNEW(I,1)      =(1.-AL)*RNEW(I,2)
             RNEW(I,M)      =AL*RNEW(I,M-1)
1       CONTINUE
             AGRID          =(IEND-IST+1.)*(M-2.)
             RMEAN          =TOTAL/AGRID
```

```
      IF(RMEAN.NE.0.0) THEN
      ERROR  =SUMDIFF/RMEAN/AGRID
      ELSE
      ERROR  =SUMDIFF/AGRID
      ENDIF
      ITNO=ITNO+1
40    IF(ERROR.LE.ERRMIN) GOTO 60
      GOTO 50
      GOTO 40
60    RETURN
      END
C     @@@@@@@-----------------------------------------------
C     FOR : FIND COEFFICIENT TERM OF R-MOMENTUM EQ.
      SUBROUTINE RCOEFT(J,M,FE,FW,FN,FS,DE,DW,DN,DS,DNP,DSP,CONTNEW,
     &                 CONTOLD,SOURCE,VROLD,AL,AP,AB)
      CALL POWLAW(FE,FW,FN,FS,DE,DW,DN,DS,ELN,ELS)
      CALL POWLAW(FE,FW,FN,FS,DE,DW,DNP,DSP,ELN,ELS)
      IF(J.EQ.2)       FW=(1.-AL)*FW
      IF(J.EQ.(M-1))   FE=AL*FE
      AE     =-FE*(1.-AL)+DE
      AW     = FW*AL+DW
      AN     =-FN*(1.-ELN)+DN
      AS     = FS*ELS+DS
      IF(J.EQ.2)       AW=0.0
      IF(J.EQ.(M-1))   AE=0.0
      AP     = AE+AW+(FE-FW)+(FN*ELN-FS*(1.-ELS))+DNP+DSP+CONTNEW
      AB     = CONTOLD*VROLD+SOURCE
      IF(J.EQ.2)       AP=AP+AL*(FW+2.*DW)
      IF(J.EQ.(M-1))   AP=AP+(1.-AL)*(2.*DE-FE)
      FE     =AE
      FW     =AW
      FN     =AN
      FS     =AS
      RETURN
      END
```

```
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C       FOR : FIND SOURCE TERM OF R- MOMENTUM EQ.
        SUBROUTINE RSOURCE(UP,UW,US,USW,VP,VN,VS,DE,DW,DN,DS,DNP,DSP
        &                 ,SOURCE)
        CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
        RSORCE=(DN*VN-DNP*VP)-(DSP*VP-DS*VS)
        ZSORCE=(DE*(UP-US)-DW*(UW-USW))*DZ/DR
        SOURCE =ZSORCE+RSORCE
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C       FOR : FIND  COEFFICIENT OF TIME TERM  OF R- MOMENTUM EQ.
        SUBROUTINE RTIME(TP,TS,DENOLD,RR,DT,VLNEW,VLOLD)
        CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
        IF(DT.NE.0.) THEN
        VLNEW =0.5*(FD(TP)+FD(TS))*RR*DZ/DT
        VLOLD =DENOLD*RR*DZ/DT
        ELSE
        VLNEW =0.0
        VLOLD =0.0
        ENDIF
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@                C
c       FOR : FIND DIFFUSTION TERM OF R- MOMENTUM EQ.
         SUBROUTINE RDIFFUS(T,N,M,I,J,RRP,RRN,RRS,RR,RN,DE,DW,DN,DS
   &              ,DNP,DSP)
        REAL T(N,M)
        CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
        CALL ATEMP(T,N,M,I,J,TP,TE,TW,TN,TS)
              VISTP           =FVIS(TP)
                     VISTS           =FVIS(TS)
                     XN                      =VISTP*DZ*RRP/DR/RRN
                     XS                      =VISTS*DZ*RRP/DR/RRS
        DE      =(VISTP+VISTS+FVIS(T(I-1,J+1))+FVIS(TE))*RR/4./DZ
```

```
      DW      =(VISTP+VISTS+FVIS(T(I-1,J-1))+FVIS(TW))*RR/4./DZ
      DN      =XN*RN
      DNP     =XN*RRP
      DS      =XS*(RRP-DR)
      DSP     =XS*RRP
      RETURN
      END
c     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR : FIND CONVECTION TERM OF R- MOMENTUM EQ.
      SUBROUTINE RCONVEC(T,UP,UW,US,USW,VP,VN,VS,RRN,RRS,RR,N,M,I,J
     &               ,FE,FW,FN,FS)
      REAL T(N,M)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      CALL ATEMP(T,N,M,I,J,TP,TE,TW,TN,TS)
      DENP    =FD(TP)
      DENS    =FD(TS)
      VPDENPS        =VP*(DENP+DENS)
      RRP            =0.5*(RRN+RRS)
      FE=RR*0.25*(US*        (DENS+FD(T(I-1,J+1)))+UP*(DENP+FD(TE)))
      FW=RR*0.25*(USW*(DENS+FD(T(I-1,J-1)))+UW*(DENP+FD(TW)))
      FN=DZ*RRP*0.25*(VPDENPS+VN*(DENP+FD(TN)))
      FS=DZ*RRP*0.25*(VPDENPS+VS*(DENS+FD(T(I-2,J))))
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR : CORRECTED VELOCITY AND PRESSURE BY ADD TO PRESSURE CORECTION
C           TERM
      SUBROUTINE CORECT(ZNEW,RNEW,PNEW,PC,APU,PRX,AL,IST,IEND,N,M)
      REAL  ZNEW(N,M-1),RNEW(N+1,M),PNEW(N,M),PC(N,M),AL
      REAL  APU(N,M-1),PRX(N+1,M)
      INTEGER IST,IEND
c     CALC.:
c           WT - WEIGHTING FACTOR OF PRESSURE
c           Z-MOMENT CORRECT :  ZNEW(:) =ZNEW(:)+APU(:)
c           R-MOMENT CORRECT :   RNEW(:) =RNEW(:)+PRX(:)
```

```
c           PRESSURE CORRECT :   PNEW(:) =PNEW(:)+PC(:)
c           CALL BCORECT FOR INCLUDE PC(:),APU(:),PRX(:) TERM
      CALL SUBCORECT(PC,APU,PRX,IST,IEND,N,M)
      WTP=0.8
      DO 1 I=IST,IEND
      DO 2 JZ=2,M-2
      ZNEW(I,JZ)      =ZNEW(I,JZ)+APU(I,JZ)
      ZNEW(I,1)       =AL*ZNEW(I,1)+(1.-AL)*ZNEW(I,2)
      ZNEW(I,M-1)     =AL*ZNEW(I,M-2)+(1.-AL)*ZNEW(I,M-1)
2     CONTINUE
      DO 3 JR=2,M-1
      RNEW(I,JR)      =RNEW(I,JR)+PRX(I,JR)
      RNEW(I,1)       =(1.-AL)*RNEW(I,2)
      RNEW(I,M)       =AL*RNEW(I,M-1)
      PNEW(I,JR)      =PNEW(I,JR)+WTP*PC(I,JR)
3     CONTINUE
1     CONTINUE
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE SUBCORECT(PC,APU,PRX,IST,IEND,N,M)
      REAL PC(N,M),APU(N,M-1),PRX(N+1,M)

c     CALC.:
c           Z-MOMENT CORRECT :  APU(:)=(PC(:)-PC(:))*APU(:)
c           R-MOMENT CORRECT :   PRX(:)=(PC(:)-PC(:))*PRX(:)
c           PRESSURE CORRECT :   PC(:) = PC(:)

      DO 1 I=IST,IEND
      DO 2 JZ=2,M-2
      APU(I,JZ)=APU(I,JZ)*(PC(I,JZ)-PC(I,JZ+1))
2     CONTINUE
      DO 3 JR=2,M-1
      PRX(I,JR)=PRX(I,JR)*(PC(I-1,JR)-PC(I,JR))
3     CONTINUE
```

```
1       CONTINUE
        RETURN
        END
C
C   SET MATRIX TO ZERO:
        SUBROUTINE GUESSZERO(PC,N,M,IST,IEND)
        REAL PC(N,M)
C   PC(:) METRIX WILL SET TO ZERO
C   N    NO. OF GRID IN RADIAL DIRECTION
C   M    NO. OF GRID IN RADIAL DIRECTION
C   IST  START GRID WILL BE SET
C   IEND  END  GRID WILL BE SET
        DO 1 I=IST,IEND
        DO 1 J=1,M
        PC(I,J)=0.0
1       CONTINUE
        RETURN
        END
C
C       FOR: CALC Z-MOMENTUM EQ.
        SUBROUTINE ZMOMT(ZNEW,RNEW,ZOLD,TNEW,TOLD,P,APU,N,M,AL,
        &                DT,IST,IEND,WT)
        REAL ZNEW(N,M-1),RNEW(N+1,M),ZOLD(N,M-1),ZGUESS(N,M-1)
        REAL TNEW(N,M),TOLD(N,M),P(N,M),APU(N,M-1)
        REAL ZMEAN,ERROR,AGRID
        REAL PW(M-1),PE(M-1),PP(M-1),PB(M-1),A(M-1),C(M-1)
                                ITNO    =0
                                FACT    =0.6
                                ERRMIN =0.00001
        CALL GUESS(ZNEW,ZGUESS,N,M-1,1,N,1.)
        CALL GUESSZERO(APU,N,M-1,1,N)
C       ------------------------------
50              SUMDIFF         =0.0
                TOTAL   =0.0
                CALL ZERO(PW,PE,PP,PB,A,C,M-1)
```

```
      DO 1 I=IST,IEND
      DO 2 J=2,M-2
            AOLD   =ZOLD(I,J)
            VNE    =RNEW(I+1,J+1)
            DENOLD       =0.5*(FD(TOLD(I,J))+FD(TOLD(I,J+1)))
            TP           =TNEW(I,J)
            TE           =TNEW(I,J+1)
      CALL RADIAL(I,RN,RS,R)
      CALL ZFLOW(ZGUESS,N,M,I,J,UP,UE,UW,UN,US)
      CALL FLOW(RNEW,N+1,M,I,J,VP,VE,VW,VN,VS)
      CALL ZDIFFUS(TNEW,AL,R,RN,RS,N,M,I,J,DE,DW,DN,DS)
      CALL ZCONVEC(TNEW,UP,UE,UW,VP,VE,VN,VNE,N,M,I,J,AE,AW,AN,AS)
      CALL ZTIME(TP,TE,DENOLD,DT,R,SNEW,SOLD)
      CALL ZSOURCE(UP,UE,UW,VP,VE,VN,VNE,DE,DW,DN,DS,SOUCE)
      CALL ZCOEFT(AE,AW,AN,AS,DE,DW,DN,DS,SNEW,SOLD,SOUCE,AOLD,AL,AP,AB)
      CALL ZFLOW(ZNEW,N,M,I,J,UP,UE,UW,UN,US)
            APU(I,J) =R/AP*WT
            WTX          =AP*(1.-WT)/WT*ZGUESS(I,J)
            CONST        =R*(P(I,J)-P(I,J+1))+WTX
      CALL TDMA1(AW,AE,AP,WT,PW(J),PE(J),PP(J),M-1,J)
      CALL TDMA2(CONST,AN,AS,AE,AW,AB,UN,US,UE,UW,PB(J),M-1,J)
      CALL TDMA3(PW(J),PE(J),PP(J),PB(J),A(J-1),C(J-1),A(J),C(J))
2     CONTINUE
      DO 10 L=M-2,2,-1
      ANS          =A(L)*ZNEW(I,L+1)+C(L)
      DIFF         =ABS(ZNEW(I,L)-ANS)
      SUMDIFF      =SUMDIFF+DIFF
      ZNEW(I,L)    =FACT*ANS+(1.-FACT)*ZNEW(I,L)
      TOTAL        =TOTAL+ABS(ZNEW(I,L))
10    CONTINUE
            ZNEW(I,1)      =AL*ZNEW(I,1)   +(1.-AL)*ZNEW(I,2)
            ZNEW(I,M-1)    =AL*ZNEW(I,M-2)         +(1.-AL)*ZNEW(I,M-1)
1     CONTINUE
            AGRID          =(IEND-IST+1.)*(M-3.)
            ZMEAN          =TOTAL/AGRID
```

```
      ERROR          =SUMDIFF/ZMEAN/AGRID
      ITNO=ITNO+1
40    IF(ERROR.LE.ERRMIN) GOTO 60
      GOTO 50
      GOTO 40
60    RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR : FIND COEFFICIENT OF Z-MOMENTUM EQ.
      SUBROUTINE ZCOEFT(FE,FW,FN,FS,DE,DW,DN,DS,CONTNEW,CONTOLD,SOURCE
      &                ,VZOLD,AL,AP,AB)
      CALL POWLAW(FE,FW,FN,FS,DE,DW,DN,DS,ELN,ELS)
      AE     =-FE*(1.-AL)+DE
      AW     = FW*AL+DW
      AN     =-FN*(1.-ELN)+DN
      AS     = FS*ELS+DS
      AP     = AE+AW+AN+AS+(FE-FW)+(FN-FS)+CONTNEW
      AB     = CONTOLD*VZOLD+SOURCE
      FE     =AE
      FW     =AW
      FN     =AN
      FS     =AS
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR : FIND SOURCE TERM OF Z-MOMENTUM EQ.
      SUBROUTINE ZSOURCE(UP,UE,UW,VP,VE,VN,VNE,DE,DW,DN,DS,SOURCE)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      ZSOURE=DE*(UE-UP)-DW*(UP-UW)
      RSOURE=(DN*(VNE-VN)-DS*(VE-VP))*DR/DZ
      SOURCE =ZSOURE+RSOURE
      RETURN
      END
```

```
C
C     FOR : FIND CONVECTION TERM OF Z-MOMENTUM EQ.
      SUBROUTINE ZCONVEC(TNEW,UP,UE,UW,VP,VE,VN,VNE,N,M,I,J,FE,FW,FN,FS)
      REAL TNEW(N,M)
      CALL RADIAL(I,RN,RS,R)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      CALL ATEMP(TNEW,N,M,I,J,TP,TE,TW,TN,TS)
      RP      =(RN+RS)/2.
      DENP    =FD(TP)
      DENE    =FD(TE)
      UPDENP=UP*(FD(TP)+FD(TE))
      FE=R*0.25*(UPDENP+UE*(DENE+FD(TNEW(I,J+2))))
      FW=R*0.25*(UPDENP+UW*(DENP+FD(TW)))
      FN=DZ*RP*0.25*(VN*(DENP+FD(TN))+VNE*(DENE+FD(TNEW(I+1,J+1))))
      FS=DZ*RP*0.25*(VP*(DENP+FD(TS))+VE*(DENE+FD(TNEW(I-1,J+1))))
      IF(I.EQ.1) FS=0.0
      RETURN
      END
C
C     FOR : FIND COEFFICIENT OF TIME TERM OF Z-MOMENTUM EQ.
      SUBROUTINE ZTIME(TP,TE,DENOLD,DT,R,VLNEW,VLOLD)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      IF(DT.NE.0.) THEN
      RDZBYDT=R*DZ/DT
      VLNEW  =0.5*(FD(TP)+FD(TE))*RDZBYDT
      VLOLD  =DENOLD*RDZBYDT
      ELSE
      VLNEW =0.0
      VLOLD =0.0
      ENDIF
      RETURN
      END
C
C     FOR : FIND DIFFUSTION TERM OF Z-MOMENTUM EQ.
      SUBROUTINE ZDIFFUS(TNEW,AL,R,RN,RS,N,M,I,J,DE,DW,DN,DS)
```

```
      REAL TNEW(N,M)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      CALL ATEMP(TNEW,N,M,I,J,TP,TE,TW,TN,TS)
      RBYDZ  =R/DZ
      DZBYDR=DZ/4./DR
      VISTP   =FVIS(TP)
      VISTE   =FVIS(TE)
      VISTPTE=VISTP+VISTE
      DE=VISTE*RBYDZ
      DW=VISTP*RBYDZ
      DN=(VISTPTE+FVIS(TN)+FVIS(TNEW(I+1,J+1)))*RN*DZBYDR
      DS=(VISTPTE+FVIS(TS)+FVIS(TNEW(I-1,J+1)))*RS*DZBYDR
C     B.C.
      IF(I.EQ.1)             DS=0.0
      IF(I.EQ.(NHIN-1))      DN=2.*DN
      IF(I.EQ.(NHOUT+1))     DS=2.*DS
      IF(I.EQ.(NCIN-1))      DN=2.*DN
      IF(J.EQ.2)             DW=AL*DW
      IF(J.EQ.(M-2))         DE=(1.-AL)*DE
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                              INPUT VALUE
C                 @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE STUDYNO(STEAD)
      REAL STEAD
      WRITE(*,*)'Select one choice for calc. '
      WRITE(*,*)'1.Calc. only steady state.'
      WRITE(*,*)'2.Calc. steady state and unsteady state continue.'
      WRITE(*,*)'Please enter choice = '
READ (*,*) STEAD
      RETURN
      END
C
```

```
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
        SUBROUTINE INPCONST(QHIN,QCIN,THIN,TCIN,AL,DT)
        REAL QHIN,QCIN,AL,DT,THIN,TCIN
C       INPUT:
C       1.HOT  FLOW RATE.(LITE/MIN)
C       2.COLD FLOW RATE.(LITE/MIN)
C       3.HOT  FLUID TEMPERATURE.(C)
C       4.COLD FLUID TEMPERATURE.(C)
C       5.INDEPENDENT WITH TIME:DT=0.0
        WRITE(*,*)'INPUT HOT FLOW RATE.(LITE/MIN)'
        READ (*,*) QHIN
WRITE(*,*)'INPUT HOT FLUID TEMPERATURE.(C)'
        READ (*,*) THIN
        WRITE(*,*)'INPUT COLD FLOW RATE.(LITE/MIN)'
        READ (*,*) QCIN
        WRITE(*,*)'INPUT COLD FLUID TEMPERATURE.(C)'
        READ (*,*) TCIN
        WRITE(*,*)'COUNTER CURRENT OR CO-CURRENT'

        WRITE(*,*)'INPUT "1" FOR CO-      CURRENT'
        WRITE(*,*)'INPUT "2" FOR COUNTER CURRENT'
        READ (*,*) AL
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
        SUBROUTINE NEWINPUT(QHIN,QCIN,THIN,TCIN,CHOICE)
        REAL QHIN,QCIN,THIN,TCIN
c   INPUT:NEW DATA THAT VARIES WITH TIME.(UNSTEADY STATE)
c    CHOICE 1:VARIES HOT TEMP.(C)
c    CHOICE 2:VARIES FLOE RATE OF HOT FLUID.(LITE/MIN)
C    CHOICE 3:VARIES FLOW RATE OF COLD FLUID.(LITE/MIN)
C    CHOICE 4:VARIES ALL CASE
        WRITE (*,*)'INPUT NEW CONDITION FOR UNSTEADY STATE.'
        WRITE (*,*)'SELECT 1 CHOICE'
        WRITE (*,*)'SELECT NO.1 FOR VARIES TEMP.OF HOT FLUID.'
```

```
      WRITE (*,*)'SELECT NO.2 FOR VARIES FLOW RATE OF HOT FLUID.'
      WRITE (*,*)'SELECT NO.3 FOR VARIES FLOW RATE OF COLD FLUID.'
      WRITE (*,*)'SELECT NO.4 FOR VARIES ALL CASE.'
      WRITE (*,*)'PLEASE ENTER CHOICE ='
      READ (*,*) CHOICE
      IF(CHOICE.EQ.1.OR .CHOICE.EQ.4.) THEN
      WRITE(*,*)'INPUT HOT FLUID TEMPERATURE.(-C)'
      READ (*,*) THIN
      ENDIF
      IF(CHOICE.EQ.2.OR .CHOICE.EQ.4.) THEN
      WRITE(*,*)'INPUT HOT FLUID RATE.(LITE/MIN)'
      READ (*,*)  QHIN
      ENDIF
      IF(CHOICE.EQ.3.OR .CHOICE.EQ.4.) THEN
      WRITE(*,*)'INPUT COLD FLUID RATE.(LITE/MIN)'
      READ (*,*)  QCIN
      ENDIF
      WRITE(*,*)'INPUT NEW COLD FLUID TEMPERATURE.(-C:MAY BE CHANGE)'
      READ (*,*)  TCIN
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C     FOR UNSTEADY STATE CASE INPUT TIME THAT WHAT KNOW SOLUTIONS .
      SUBROUTINE ATTIME(TIMEEND)
      REAL TIMEEND
      WRITE(*,*)'INPUT TIME THAT WANT TO KNOW RESULT:(SEC)'
      READ (*,*)  TIMEEND
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE THOTCHAGE(TNEW,THNEW,TCNEW,AL,N,M)
      REAL TNEW(N,M)
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      DO 1 I=1,NHIN-1
      TNEW(I,1)=THNEW
```

```
1       CONTINUE
C       DO 2 IC=NHOUT+1,NCIN-1
C       TNEW(IC,1)=AL*TCNEW+(1.-AL)*TNEW(IC,1)
C       TNEW(IC,M)=AL*TNEW(IC,M)+(1.-AL)*TCNEW
C2   CONTINUE
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
        SUBROUTINE HOTVZNEW(QHNEW,QHOLD,TIMESTEP,DT,VZIN)
        REAL QHNEW,QHOLD,TIMESTEP,DT,VZIN
        CALL HOTVELO(QHNEW,VHNEW)
        CALL HOTVELO(QHOLD,VHOLD)
        CALL VARIEFLOW(VHNEW,VHOLD,TIMESTEP,DT,DIFFVELO)
        VZIN=VZIN+(VHNEW-VHOLD)/(TIMESTEP/DT)
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
        SUBROUTINE VARIEFLOW(VZNEW,VZOLD,TIMESTEP,DT,DIFFVELO)
        REAL VZNEW,VZOLD,TIMESTEP,DT,DIFFVELO
C       CALC:
C        1. VELOCITY CHANGE IN TIME STEP :VELOCITY/TIMESTEP
C        2. VELOCITY CHANGE PER ITERATIVE:VELOCITY/(TIMESTEP/DT)
        DIFFVELO=(VZNEW-VZOLD)/(TIMESTEP/DT)
        RETURN
        END
C       @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
        SUBROUTINE TCDCHAGE(TNEW,N,M,TCNEW,AL)
        REAL TNEW(N,M)
        CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
        DO 10 I=NHOUT+1,NCIN-1
        TNEW(I,1)=AL*TCNEW+(1.-AL)*TNEW(I,1)
        TNEW(I,M)=AL*TNEW(I,M)+(1.-AL)*TCNEW
10      CONTINUE
        RETURN
        END
```

```
C          @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
           SUBROUTINE COLDVZNEW(QCNEW,QCOLD,TIMESTEP,DT,VZIN)
           REAL QCNEW,QCOLD,TIMESTEP,DT,VZIN
           CALL COLDVELO(QCNEW,VCNEW)
           CALL COLDVELO(QCOLD,VCOLD)
           CALL VARIEFLOW(VCNEW,VCOLD,TIMESTEP,DT,DIFFVELO)
           VZIN=VZIN+DIFFVELO
           RETURN
           END
C          @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
           SUBROUTINE HOTVELO(QHIN,VHIN)
           REAL QHIN,VHIN,RADIALHOT,CROSSAREAHOT,CONVERSEQHIN
c          CALC.:
c          HOT VELOCITY INPUT=(HOT FLOW RATE /CROSSECTIONAL AREA )
           CALL RPIPE(RADIALHOT,RCOLDOUT,RCOLDIN,RATIO)
           CROSSAREAHOT=3.1415927*RADIALHOT**2.
           CONVERSEQHIN=QHIN/60./1000.
           VHIN          =CONVERSEQHIN/CROSSAREAHOT
           RETURN
           END
C          @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
           SUBROUTINE COLDVELO(QCIN,VCIN)
           REAL QCIN,VCIN,RCOLDOUT,RCOLDIN,CROSSAREACOLD,CONVERSEQCIN
c          CALC.:
c          COLD VELOCITY INPUT=(COLD FLOW RATE /CROSSECTIONAL AREA )
           CALL RPIPE(RADIALHOT,RCOLDOUT,RCOLDIN,RATIO)
           CROSSAREACOLD=3.1415927*(RCOLDOUT**2.-RCOLDIN**2.)
           CONVERSEQCIN  =QCIN/60./1000.
           VCIN          =CONVERSEQCIN/CROSSAREACOLD
           RETURN
           END
C          @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                         CONSTANT PARAMETER
C                 @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
           SUBROUTINE DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
```

```
      REAL DR,DZ
      INTEGER     NHIN,NHOUT,NCIN,NCOUT
c      CONSTANT:
c          DZ         =AXIAL  DIFFERENCE     (m)
c          DR         =RADIAL DIFFERENCE     (m)
c          NHIN   =NO.CELL AT INNER DIAMETER TUBE:(HOT PIPE)
c          NHOUT=NO.CELL AT OUTER DIAMETER TUBE:(HOT PIPE)
c          NCIN   =NO.CELL AT INNER DIAMETER SHELL:(COLD PIPE)
c          NCOUT=NO.CELL AT OUTER DIAMETER SHELL:(COLD PIPE)
      DZ     =0.05
      DR     =0.0005
      NHIN   =16
      NHOUT  =17
      NCIN   =25
      NCOUT  =26
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE RPIPE(RADIALHOT,RCOLDOUT,RCOLDIN,RATIO)
      REAL RADIALHOT,RCOLDOUT,RCOLDIN,RATIO
C     CALC:
C     RADIAL OF HOT TUBE AND COLD SHELL.
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      RADIALHOT    =DR*(NHIN-1.)
      RCOLDOUT     =DR*(NCIN-1.)
      RCOLDIN      =DR*NHOUT
      RATIO        =RCOLDIN/RCOLDOUT
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                          READ MATRIX
C          @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE ATEMP(T,N,M,I,J,TP,TE,TW,TN,TS)
      REAL T(N,M)
      TP     =T(I,J)
```

```
      TE      =T(I,J+1)
      TW      =T(I,J-1)
      TN      =T(I+1,J)
      TS      =T(I-1,J)
      IF(I.EQ.1) TS=TP
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE ZFLOW(VZ,N,M,I,J,UP,UE,UW,UN,US)
      REAL VZ(N,M-1)
      UP=VZ(I,J)
      UE=VZ(I,J+1)
      UW=VZ(I,J-1)
      UN=VZ(I+1,J)
      US=VZ(I-1,J)
      IF(I.EQ.1) US=UP
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE FLOW(A,N,M,I,J,UP,UE,UW,UN,US)
      REAL A(N,M)
      UP=A(I,J)
      UE=A(I,J+1)
      UW=A(I,J-1)
      UN=A(I+1,J)
      US=A(I-1,J)
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                         WRITE RESULT
C           @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE WUNSTED(ZNEW,RNEW,PNEW,TNEW,N,M)
      REAL ZNEW(N,M-1),RNEW(N+1,M),PNEW(N,M),TNEW(N,M)
C   WRITE:
C       1.Z-VELOCITY(M/SEC)
```

```
C          2.R-VELOCITY(M/SEC)
C         3.PRESSURE (KG/(M*SEC^2.))
C          4.TEMPERATURE(C)
          WRITE(111,1000)((ZNEW(I,J),I=1,N),J=1,M-1)
          WRITE(112,2000)((RNEW(I,J),I=1,N+1),J=1,M)
          WRITE(113,3000)((PNEW(I,J),I=1,N),J=1,M)
          WRITE(114,4000)((TNEW(I,J),I=1,N),J=1,M)
1000      FORMAT(2X,'Z FLOW',/26(1X,F7.5))
2000      FORMAT(2X,'R FLOW',/27(1X,F8.6))
3000      FORMAT(2X,'PRESS ',/26(1X,F10.5))
4000      FORMAT(2X,'TEMP  ',/26(1X,F6.3))
          RETURN
          END
C         @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
          SUBROUTINE WSTEADY(ZNEW,RNEW,PNEW,TNEW,N,M)
          REAL ZNEW(N,M-1),RNEW(N+1,M),PNEW(N,M),TNEW(N,M)
C   WRITE:
C          1.Z-VELOCITY(M/SEC)
C          2.R-VELOCITY(M/SEC)
C         3.PRESSURE (KG/(M*SEC^2.))
C          4.TEMPERATURE(C)
          WRITE(1,1000)((ZNEW(I,J),I=1,N),J=1,M-1)
          WRITE(2,2000)((RNEW(I,J),I=1,N+1),J=1,M)
          WRITE(3,3000)((PNEW(I,J),I=1,N),J=1,M)
          WRITE(4,4000)((TNEW(I,J),I=1,N),J=1,M)
1000      FORMAT(2X,'Z FLOW',/26(1X,F7.5))
2000      FORMAT(2X,'R FLOW',/27(1X,F8.6))
3000      FORMAT(2X,'PRESS ',/26(1X,F10.5))
4000      FORMAT(2X,'TEMP  ',/26(1X,F6.3))
          RETURN
          END
C         @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
          SUBROUTINE GUESSTNEW(THIN,TCIN,TNEW,N,M)
          REAL TNEW(N,M),THIN,TCIN
C         GUESS:
```

```
C     TEMPERATURE AT FRIST CALC.
C     TEMP. OF FLUID IN TUBE EQUAL THIN
C     TEMP. OF SOLID AND FLUID IN SHELL EQUAL TCIN
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      DO 1 J=1,M
      DO 2 IH=1,NHIN-1
      TNEW(IH,J)=THIN
2     CONTINUE
      DO 3 I=NHIN,NCOUT
      TNEW(I,J) =TCIN
3     CONTINUE
1     CONTINUE
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE GUESSPNEW(PNEW,VHIN,THIN,VCIN,TCIN,AL,N,M)
      REAL PNEW(N,M),ALENGHT,ANT,APLUS,BPLUS,AL
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      CALL RPIPE(RADIALHOT,RCOLDOUT,RCOLDIN,RATIO)
      ALENGHT=DZ*M
      BAT      =1.-RATIO**2.
      CPLUS    =(1.-RATIO**4.)/BAT-BAT/log(1./RATIO)
      DO 1 J=1,M
      DO  2 I=1,N
          PNEW(I,J)=0.0
2     CONTINUE
          ANT      =(J-1.)/(M-1.)
          APLUS    =1.-ANT
          BPLUS    =AI*(1.-2.*ANT)+ANT
      DO 10 IH=1,NHIN-1
          CAR            =VHIN*8.*FVIS(THIN)*ALENGHT/RADIALHOT **2
          PNEW(IH,J)     =CAR*APLUS-CAR
10    CONTINUE
```

```
      DO 20 IC=NHOUT+1,NCIN-1
            FAN              =VCIN*8.*FVIS(TCIN)*ALENGHT/CPLUS/RCOLDOUT**2.
            BAN              =FAN*BPLUS
            PNEW(IC,J)    =BAN-FAN
20    CONTINUE
1   CONTINUE
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE GUESSRNEW(RNEW,N,M)
      REAL RNEW(N+1,M)
C     GUESS:
C   PRESSURE AT FRIST CALC.(RNEW(:))
            DO 1 IA=1,N
            DO 1 JA=1,M
            RNEW(IA,JA)=0.0
1           CONTINUE
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE GUESSZNEW(ZNEW,VHIN,VCIN,AL,N,M)
      REAL ZNEW(N,M-1),AL
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      CALL RPIPE(RADIALHOT,RCOLDOUT,RCOLDIN,RATIO)
C     GUESS:
C      1.HOT TUBE
C      ASSUME : PARABOLIC VELOCITY PROFILE ALONG THE TUBE EXCEPT Z=0.0(INPUT)
C      DETERMINE: VELOCITY INPUT IN HOT PIPE EQUAL ALL CROSSECTION.
C     2.COLD SHELL
C     DETERMINE: VELOCITY INPUT IN COLD PIPE EQUAL ALL CROSSECTION.
      DO 1 J=1,M-1
      DO 2 IA=1,N
      ZNEW(IA,J)=0.0
2     CONTINUE
      DO 3 IB=1,NHIN-1
```

```
          CALL RADIAL(IB,RN,RS,R)
          ZNEW(IB,J)=2.*VHIN*(1.-((RN+RS)/2./DR/(NHIN-1))**2.)
          ZNEW(IB,1)=VHIN
3    CONTINUE
      DO 4 IC=NHOUT+1,NCIN-1
          CALL RADIAL(IC,RN,RS,R)
          SRATIO=0.5*(RN+RS)/RCOLDOUT
          FAT     =1.-SRATIO**2.+(1.-RATIO**2.)/LOG(1./RATIO)*LOG(SRATIO)
          BAT     =1.-RATIO**2.
          CPLUS   =(1.-RATIO**4.)/BAT-BAT/log(1./RATIO)
          APPLE   =FAT/CPLUS
          COOK    =2.*VCIN*APPLE
          ZNEW(IC,J)     =COOK*AL+(AL-1)*COOK
          ZNEW(IC,1)     =AL*VCIN+(1.-AL)*ZNEW(IC,1)
          ZNEW(IC,M-1)=AL*ZNEW(IC,M-1)+(AL-1.)*VCIN
4     CONTINUE
1     CONTINUE
      RETURN
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
      SUBROUTINE GUESS(ANEW,AOLD,NROW,MCOL,IST,IEND,UNSTEADY)
      REAL ANEW(NROW,MCOL),AOLD(NROW,MCOL),UNSTEADY
C   MEAN:
C   NROW        :NO.ROW OF A(:)
C   MCOL        :NO.COLUMN OF A(:)
C   IST         :NO.ROW START CALC.
C   IEN         :NO.ROW END CALC.
C   UNSTEADY:
C        1.UNSTEADY =0.0 :CALC.STEADY STATE SO
C        AOLD(:) EQUAL ZERO AT STEADY STATE
C        2.UNSTEADY =1.0 :CALC.UNSTEADY STATE SO
C        AOLD(:) EQUAL ANEW(:) AT PREVIOUS TIME
      IF(UNSTEADY.EQ.0.0)THEN
          DO 1 IA=IST,IEND
          DO 1 JA=1,MCOL
```

```
                    AOLD(IA,JA)=0.0
                    AOLD(IA,JA)=0.0
1                   CONTINUE
          ELSE
                    DO 10 IB=IST,IEND
                    DO 10 JB=1,MCOL
                    AOLD(IB,JB)=ANEW(IB,JB)
                    AOLD(IB,JB)=ANEW(IB,JB)
10        CONTINUE
          ENDIF
          RETURN
          END
C         @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
          SUBROUTINE ZERO(PW,PE,PP,PB,A,B,LDX)
          REAL PW(LDX),PE(LDX),PP(LDX),PB(LDX),A(LDX),B(LDX)
C         SET :
C         TRANSFER MATRIX IN TDMA ALGORITHM EQUAL ZERO
          DO 1 I=1,LDX
          PW(I)=0.0
          PE(I)=0.0
          PP(I)=0.0
          PB(I)=0.0
          A(I)=0.0
           B(I)=0.0
1         CONTINUE
          RETURN
          END
C         @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
          SUBROUTINE RADIAL(I,RN,RS,R)
          REAL  RN,RS,R
C         DETERMINE:
C         RS,RRS :INTERFACE OF CELL IN SOUTH DIRECTION
C         RN,RRN :INTERFACE OF CELL IN NORTH DIRECTION
C         R,RR   :DR*RCELL =DR*((RN+RS)/2.)
C         DISTANCE FOR CENTER LINE OF GRID PIONT OF Z-VELOCIY,TEMP,PRESSURE
```

```
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      RS      =DR*(I-1)
      RN      =DR*I
      R       =0.5*(RN**2.-RS**2.)
      RETURN
      END
C     [illegible]@@@@@@@@@@@@@
      SUBROUTINE RRADIA(IR,RRN,RRS,RR)
C     DISTANCE FOR CENTER LINE OF GRID PIONT OF R-VELOCIY
      CALL DHEXMETER(NHIN,NHOUT,NCIN,NCOUT,DR,DZ)
      RRS =DR*(IR-1)-0.5*DR
      RRN =DR*IR-0.5*DR
      RR      =0.5*(RRN**2.-RRS**2.)
      RETURN
      END
```

```
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                         LIQUID PROPERTY
C           @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C      1.HEAT CAPACITY               :FCP (W/KG/K)
C      2.HEAT CONDUCTION             :FK   (W/M/K )
C      3.DENSITY                     :FD  (KG/M^3)
C      4.VISCOSITY                   :FVIS(KG/M/SEC)

1     FUNCTION FCP(T)
      REAL T,FCP
      FCP=((7.E-06)*(T+273.)**2.-0.0045*(T+273.)+4.8558 )*1000
      END
2     FUNCTION FK(T)
      REAL T,FK
      FK=2*(-0.0075*(T+273.)**2.+5.9536*(T+273.)-479.14)/1000.
      END
3     FUNCTION FD(T)
      REAL FD,T
      FD=-0.0039*(T+273.)**2.+2.0583*(T+273.)+726.15
      END
4     FUNCTION FVIS(T)
      REAL T,FVIS
      FVIS=(0.2162*(T+273.)**2.-151.88*(T+273.)+27011.)*1.E-6
      END
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C                    SOLID PROPERTY :Rhodium
C     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
C      1.HEAT CAPACITY               :FCPS (W/KG/K)
C      2.HEAT CONDUCTION             :FKS (W/M/K )
C      3.DENSITY                     :FDS  (KG/M^3)
1   FUNCTION FCPS(T)
      FCPS=Acps+(T-T)
      FCPS=0.0878*T+273.93
      END
2   FUNCTION FKS(T)
```

```
      AKS=149.0
      FKS=Aks+(T-T)
      END
3     FUNCTION FDS(T)
      ADS=12.4*1000
      FDS=Ads+(T-T)
      END
```

2. โปรแกรม Matlab สำหรับแสดงผล

2.1 โปรแกรมแสดงกราฟของอุณหภูมิ

```
load D:\Data\Temp.dat
t=Temp;
Pcolor(t);
Shading flat
```

2.3 โปรแกรมแสดงกราฟของความเร็ว

```
load D:\Data\Vz.dat
z=Vz;
colormap(cool);
Pcolor(z);
Hold on
Surf(z)
Hold off
```

ภาคผนวก ฉ

Notation

A	=	area, m^2
C_p	=	heat capacity at constant pressure, per unit mass ,$m^2/s^2\ ^\circ C$
D	=	diameter of cylinder, m
E	=	$i + K + \Phi$ = total fluid energy, $kg.m^2/sec^2$
g	=	gravitational acceleration ,m^2/sec
i	=	internal energy, $kg.m^2/sec^2$
K	=	kinetic energy, $kg.m^2/sec^2$
k	=	thermal conductivity, $kg.m/sec^3\ ^\circ C$
p	=	fluid pressure, $kg/m.sec^2$
P	=	$p \pm \rho gh$, $kg/m.sec^2$
q	=	energy fluxes, kg/sec^3
r	=	radius distance in cylindrical coordinates, m
S_E, S_i	=	energy source, $kg/m.sec^3$
T	=	temperature, $^\circ C$
t	=	time, sec
u	=	axial velocity of fluid in cylindrical coordinates, m
v	=	direction velocity of fluid in cylindrical coordinates, m
w	=	angular velocity of fluid in cylindrical coordinates, m
z	=	axial distance in cylindrical coordinates, m
μ	=	viscosity, kg/ m.sec
ρ	=	fluid density, kg/m^3
θ	=	angle in cylindrical, radians
τ	=	viscosity stress tensor, $kg/m.sec^2$
Φ	=	potential energy, $kg.m^2/sec^2$
Ψ	=	viscous dissipation, sec^{-2}

Superscripts

′	=	correction
*	=	guess
o	=	old value

Commonly Used Dimemsionless group

Nu	=	Nusselt number for heat transfer
Pe	=	Peclet number
Pr	=	Prandtl number
Re	=	Reynolds number

Mathematical Operation

D/Dt	=	substantial derivative

ประวัติผู้เขียน

นางสาวอรพรรณ บุญช่วย เกิดวันที่ 18 กันยายน 2517 จังหวัดกรุงเทพฯ สำเร็จการศึกษาปริญญาตรีวิทยาศาสตร์บัณฑิต สาขาเคมีวิศวกรรม(เทคโนโลยีทางเชื้อเพลิง) ภาควิชาเคมีเทคนิค คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2538 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตร์มหาบัณฑิต ที่จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2539