

ภาคผนวก ก.

ก.1 แบบจำลองที่ใช้คำนวณหาปริมาณน้ำท่า

แบบจำลองที่ใช้ในการหาปริมาณน้ำท่าสำหรับพื้นที่รับน้ำที่ไม่มีสถานีวัดน้ำ จะใช้แบบจำลองที่พัฒนาโดย นาถฤทธิ์ (2529)* ซึ่งเป็นแบบจำลองเพื่อหาน้ำท่าโดยขึ้นอยู่กับลักษณะของดินและการใช้พื้นที่รวมทั้งเงื่อนโซ่ ความชันก่อนหน้าด้วย แบบจำลองดังกล่าวจะใช้ทำนายความสัมพันธ์ระหว่างน้ำฝนกับน้ำท่าของกลุ่มน้ำที่พิจารณา ดูรายละเอียดในภาคผนวก ข.

ก.2 การคำนวณหาความต้องการใช้น้ำ

ก.2.1 การคำนวณการใช้น้ำของพืชอ้างอิง

โดยใช้วิธีของ Pan Evaporation method ซึ่งเป็นการคำนวณจากค่าการระเหยของอากาศ ในการทดลองหาค่าการระเหยของพืช มักจะหาค่าความสัมพันธ์กับการระเหยจากอากาศ Class A ต่อมาได้ปรับปรุงแก้ไขโดย Doorenbos & Pruitt (1977) และสามารถเขียนความสัมพันธ์ได้ว่า

$$ET_p = K_p \quad E_{pan} \dots\dots\dots ก.1$$

โดย ET_p = ค่าการคายระเหยของพืชอ้างอิง มม./วัน
 K_p = สัมประสิทธิ์ของอากาศ ดูจากตารางที่ ก.1
 E_{pan} = ค่าการระเหยของอากาศ มม./วัน

* นาถฤทธิ์ ผดุงศรี (2532) น้ำท่าจากลุ่มน้ำที่มีลักษณะแปรผันไปหาการใช้พื้นที่, ชนิดของดิน และลักษณะภูมิประเทศ ; วิทยานิพนธ์มหาบัณฑิต มหาวิทยาลัยขอนแก่น : ขอนแก่น

ตารางที่ ก.1 ค่า K_p สำหรับคุณค่าการคายระเหยจากผาตเพื่อให้ได้ค่าศักยภาพการคายระเหยของพืชอ้างอิง

อ่างชั้น A	กรณี ก. อ่างล้อมรอบด้วยพืชเขียวสั่น			กรณี ข. อ่างล้อมรอบด้วยที่ว่างแห้ง			
ความเร็ว ลม กม./วัน	ความชื้นสัมพัทธ์เฉลี่ย (%)			ระยะลมพัด ก่อนถึงอ่าง (เมตร)	ความชื้นสัมพัทธ์เฉลี่ย (%)		
	ต่ำ <40	ปานกลาง 40-70	สูง >70		ต่ำ <40	ปานกลาง 40-70	สูง >70
อ่อน <170	0.55	0.65	0.75	0	0.70	0.80	0.85
	0.65	0.75	0.85	10	0.60	0.70	0.80
	0.70	0.80	0.85	100	0.55	0.65	0.75
	0.75	0.85	0.85	1000	0.50	0.60	0.70
ปานกลาง 170-425	0.50	0.60	0.65	0	0.65	0.75	0.80
	0.60	0.70	0.75	10	0.55	0.65	0.70
	0.65	0.75	0.80	100	0.50	0.60	0.65
	0.70	0.80	0.80	1000	0.45	0.55	0.60
แรง 425-700	0.45	0.50	0.60	0	0.60	0.65	0.70
	0.55	0.60	0.65	10	0.50	0.55	0.65
	0.60	0.65	0.70	100	0.45	0.50	0.60
	0.65	0.70	0.75	1000	0.40	0.45	0.55
แรงมาก >700	0.40	0.45	0.50	0	0.50	0.60	0.65
	0.45	0.55	0.60	10	0.45	0.50	0.55
	0.50	0.60	0.65	100	0.40	0.45	0.50
	0.55	0.60	0.65	1000	0.35	0.40	0.45

หมายเหตุ ในกรณี ข. ถ้าพื้นที่ว่างดินแห้งมีอาณาเขตกว้างอยู่ในลักษณะเช่นนี้ เรื่อยไปจะลดค่า K_p ได้ 20 % ในสภาพลมร้อน และลดลงได้ 5-10 % ในกรณีของความชื้น อุณหภูมิและกระแสน้ำมีค่าปานกลาง

ก.2.2 ค่าสัมประสิทธิ์การใช้น้ำเปรียบเทียบกับพืชอ้างอิง

ตามปกติแล้วพืชจะมีการคายน้ำในปริมาณต่าง ๆ กันโดยขึ้นอยู่กับช่วงระยะเวลาของการเจริญเติบโตถ้าเป็นระยะต้น ๆ จะมีค่าน้อยและมากที่เมื่อเจริญเติบโตเต็มที่และลดลงเมื่อพืชออกผล ซึ่งการเจริญเติบโตนี้สามารถแบ่งออกได้ 3 ระยะคือช่วงแตกใบ (Vegetative stage) ช่วงออกดอก (Flowing stage) และช่วงออกผล (Fruiting stage) ดังเห็นการใช้น้ำของพืช

$$ET = k ET_p \dots\dots\dots ก.2$$

โดย ET = ค่าการคายระเหยของพืช มม./วัน

k = ค่าสัมประสิทธิ์การใช้น้ำเปรียบเทียบกับพืชอ้างอิง (ดูตารางที่ ก.2)

ET_p = ค่าการคายระเหยของพืชอ้างอิง มม./วัน

ตารางที่ ก.2 ค่าสัมประสิทธิ์การใช้น้ำเปรียบเทียบกับพืชอ้างอิง

เดือน	อ้อย	ผัก	พืชล้มลุก	พืชไร่	เดือนหลังจากปักดำ	ข้าวพันธุ์เก่าเมือง	ข้าวพันธุ์ กข.
มกราคม	0.63	0.67	0.50	0.40	1	0.83	1.00
กุมภาพันธ์	0.83	0.67	0.80	0.70	2	0.93	1.12
มีนาคม	1.00	0.67	0.70	1.00	3	1.06	1.27
เมษายน	1.13	0.67		0.80	4	0.96	1.18
พฤษภาคม	1.18	0.67		0.30	5	0.98	
มิถุนายน	1.18	0.67					
กรกฎาคม	1.13	0.67					
สิงหาคม	1.03	0.67					
กันยายน	0.85	0.67					
ตุลาคม	0.65	0.67					
พฤศจิกายน	0.53	0.67					
ธันวาคม	0.50	0.67					

ก.2.3 ค่าชลภาวะของพืช

หลักเกณฑ์ในการประเมินหาความต้องการน้ำชลประทาน พอจะสรุปเป็นสูตรได้ดังนี้

$$\text{ค่าชลภาวะของพืช} = \frac{\text{ความลึกของน้ำชลประทาน} \times 1600}{1000} \text{ (ม}^3\text{/ไร่/ไร่)...ก.3}$$

$$\text{เมื่อ ความลึกของน้ำชลประทาน} = ET + P - ER \text{ (มม./วัน)...ก.4}$$

ET = ค่าการคายระเหยของพืช (Evapotranspiration) มม./วัน

P = อัตราการรั่วซึมของน้ำในแปลง (Percolation & Seepage) มม./วัน

ER = ปริมาณฝนที่ใช้การได้ (Effective Rainfall) มม./วัน

สำหรับการหาปริมาณของฝนที่ใช้การ ใช้สูตรของ เอฟ เอ โอ (FAO Formulas)

$$ER = (1 - 0.006 R) \times R \text{ก.5}$$

$$\text{เมื่อ } ER = \text{ปริมาณฝนที่ใช้การได้} \quad \text{มม./วัน}$$

$$R = \text{ปริมาณฝนที่ตก} \quad \text{มม./วัน}$$

ก.2.4 การคำนวณปริมาณน้ำที่ส่งเพื่อการชลประทาน

$$\text{SUM1} = \frac{\text{ค่าชลภาวะของพืช} \times \text{พื้นที่เพาะปลูก}}{\text{ประสิทธิภาพการชลประทาน}}$$

จากนั้นคำนวณหาปริมาณน้ำที่ใช้เพื่อการเพาะปลูกสะสมรายวัน

$$\text{SUM2} = \text{SUM2} + \text{SUM1} \text{ก.6}$$

$$\text{เมื่อ SUM2} = \text{ปริมาณน้ำที่พืชต้องใช้เพื่อการเพาะปลูกสะสมรายวัน (ล.บ.ม ต่อ วัน)}$$

$$\text{ค่าชลภาวะของพืช} = \text{มีหน่วย ล.บ.ม ต่อวันต่อไร่}$$

$$\text{พื้นที่เพาะปลูก} = \text{มีหน่วย ไร่}$$

$$\text{ประสิทธิภาพของการชลประทาน} = 0.65$$

ก.3 การวิเคราะห์เสถียรภาพของตัวเขื่อน

A.W. Bishop ได้วิเคราะห์หาเสถียรภาพความลาดของวิธีแบ่งดินที่อยู่บนเขื่อนระบายน้ำเคลื่อนตัวออกเป็นส่วนๆ (method of slice) จากการพิจารณาแรงกระทำของดินแต่ละส่วนที่แบ่งดังรูปที่ ก.3 โดยที่เสถียรภาพองค์ประกอบของความลาดขึ้นอยู่กับค่าปัจจัยความปลอดภัยที่สามารถต้านทานการบิด ซึ่งมีความเท่ากับอัตราส่วนของกำลังหน่วยแรงเฉือนสูงสุดที่ดินเกิดการเคลื่อนตัว ($C_{available}$) เทียบกับความต้านทานของหน่วยแรงเฉือนที่ต้องการ สำหรับสภาพสมมูลย์ ($C_{mobilized}$)

$$F.S. = \frac{\tau_{available}}{\tau_{mobilized}} \dots\dots\dots \text{ก.7}$$

หน่วยแรงประสิทธิผล

$$\tau_{available} = c' + (\sigma' - u) \tan \phi' \dots\dots\dots \text{ก.8}$$

$$\tau_{mobilized} = 1[c' + (\sigma' - u) \tan \phi'] \dots\dots\dots \text{ก.9}$$

โดยที่

F.S.

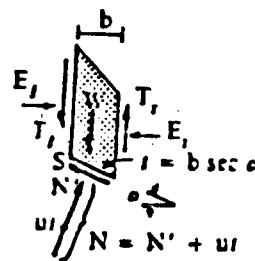
$\tau_{available}$ = กำลังหน่วยแรงเฉือนสูงสุดที่ดินเกิดการเคลื่อนตัว

$\tau_{mobilized}$ = ความต้านทานของหน่วยแรงเฉือนที่ต้องการสำหรับสภาพสมดุล

c' = แรงแยัดเหนียวหรือแรงเชื่อมแน่นประสิทธิผล

σ' = หน่วยแรงกดหรือแรงตั้งฉากประสิทธิผลบนระนาบแรงเฉือน

ϕ' = มุมเสียดทานภายในหรือมุมต้านทานต่อแรงเฉือนประสิทธิผล



รูปที่ ก.1 ส่วนของความลาดกับแรงต่างๆ ที่กระทำในการวิเคราะห์ของ A.W.Bishop จากรูปที่ ก.1

W = น้ำหนักของดิน

N = แรงรวมตั้งฉากกับพื้นระนาบเคลื่อนตัว

N' = แรงประสิทธิผลตั้งฉากกับพื้นระนาบเคลื่อนตัว

ul = แรงเนื่องจากความดันของน้ำ

u = ความดันของน้ำที่กระทำใต้ฐานของดินส่วนที่แบ่ง ซึ่งมีค่าเท่ากับ $r_w Z_w$

r_w = หน่วยน้ำหนักของน้ำ

Z_w = ระยะความลึกจากระดับผิวน้ำ

E_1, E_2 = แรงตั้งฉากบนพื้นระนาบตั้งแนวส่วนแบ่งของดิน

T_1, T_2 = แรงเฉือนบนพื้นระนาบตั้งแนวส่วนแบ่งของดิน

ในการวิเคราะห์หน่วยแรงประสิทธิผลของเสถียรภาพความลาดด้วยวิธี A.W.Bishop พบ

$$W = N' \cos \alpha + ul \cos \alpha + (T_r - T_l) + \frac{1}{F.S.} [c' + (N - ul) \tan \phi'] \sin \alpha$$

$$W = N' \cos \alpha + ul \cos \alpha + (T_r - T_l) + \frac{c' l \sin \alpha}{F.S.} + \frac{(N - ul) \tan \phi' \sin \alpha}{F.S.}$$

$$\text{แต่ } b = l \cos \alpha$$

$$N' = \frac{W - ub - \left[\frac{c' l \sin \alpha}{(F.S.)} \right] - (T_r - T_l)}{\cos \alpha + \left[\frac{\tan \phi' \sin \alpha}{F.S.} \right]} \quad \dots\dots\dots \text{ก. 10}$$

$$F.S. = \frac{\sum \left[c' l + \tan \phi' \left[\frac{W - ub - \left(\frac{c' l \sin \alpha}{F.S.} \right) - (T_r - T_l)}{\cos \alpha + \frac{\tan \phi' \sin \alpha}{F.S.}} \right] \right]}{\sum W \sin \alpha} \quad \dots\dots\dots \text{ก. 11}$$

ก.4 สภาพของเขื่อนที่ทำการวิเคราะห์เสถียรภาพความลาด

สภาพของเขื่อนที่ทำการวิเคราะห์มี 3 สภาพคือ

ก. หลังเสร็จสิ้นการก่อสร้าง (End of Construction) ซึ่งในการบดอัดจะมีการพรมน้ำเพื่อให้ได้ความชื้นที่เหมาะสม หลังเสร็จสิ้นการก่อสร้างความดัน Pore ส่วนหนึ่งจะสูญเสียไปขึ้นอยู่กับชนิดของดินและเทคนิคการก่อสร้าง และน้ำใต้ดินที่น้ำเขื่อนยังไม่มีที่จะช่วยดันผิวทางด้านเหนือน้ำการวิเคราะห์สภาพนี้หาได้ซึ่งความปลอดภัยทั้งของทางด้านหน้าเขื่อนและทางด้านท้ายเขื่อน

ข. น้ำหน้าเขื่อนลดระดับอย่างรวดเร็ว (Rapid drawdown) ในสภาพนี้ถ้าในเขื่อนจะถูกระบายออกในอัตราหนึ่ง ทำให้ระดับน้ำหน้าเขื่อนลดลงอย่างรวดเร็วเป็นเหตุให้ความดัน Pore ของเขื่อนทางด้านเหนือน้ำสลายออกไม่เร็วพอ เป็นเหตุให้แรงกระทำค่อนข้างสูงในขณะที่แรงช่วยรับจากระดับน้ำหน้าเขื่อนไม่มี ในกรณีนี้การวิเคราะห์ทำเฉพาะผิวด้านหน้าเขื่อนเท่านั้น

ค. สภาพไหลซึมคงตัว (Steady State) ในสภาพนี้ถือว่าระดับน้ำหน้าเขื่อนที่ระดับเก็บกักคงที่ระดับนั้นนานพอสมควรที่การไหลซึมผ่านตัวเขื่อนอยู่ในสภาพคงตัว กรณีนี้แรงกระทำต่อผิวตัวเขื่อนทางด้านท้ายเขื่อนยังคงเดิมแต่แรงรับน้อยลง เนื่องจากสภาพอิ่มตัวบางส่วนที่เกิดขึ้นการวิเคราะห์เสถียรภาพจะทำเฉพาะทางด้านท้ายเขื่อนเท่านั้น

ค่าดัชนีความปลอดภัยของเขื่อนดินน้อยที่สุดจากการวิเคราะห์สำหรับเขื่อนดินในสภาพต่างๆ กันควรมีค่าไม่น้อยกว่าค่าในตารางที่ ก.3

ตารางที่ ก.3 ค่าดัชนีความปลอดภัยของเขื่อนดินน้อยที่สุด (USBR, 1974)

สถานการณ์ใช้งาน	ดัชนีความปลอดภัย	
	ด้านท้ายเขื่อน	ด้านท้ายเขื่อน
1. หลังเสร็จสิ้นการก่อสร้าง	1.25	1.25
2. น้ำหน้าเขื่อนลดระดับอย่างรวดเร็ว	1.25	—
3. การไหลซึมคงตัว	—	1.25

ก.5 การหาปริมาณน้ำนอง(Flood peak)

นอกจากปริมาณน้ำนองสูงสุด (Flood Peak) จะมีความสำคัญต่อความกว้างสันทางระบายน้ำแล้ว ปริมาณของน้ำท่า (Run off Volume) ก็มีส่วนสำคัญมากด้วยโดยเฉพาะอย่างยิ่ง การหาความกว้างของทางระบายน้ำล้นของอ่างเก็บน้ำ ซึ่งมีความสามารถลดและชะลอการเกิดอัตราการไหลออกสูงสุด (Attenuation and delaying effect) เมื่อผ่านทางระบายน้ำล้น ซึ่งอัตราการไหลออกสูงสุดจะลดลงมากหรือน้อยแค่ไหนขึ้นอยู่กับปริมาณการเก็บกักของอ่างเก็บน้ำเหนือทางระบายน้ำล้น (Surcharge Storage)

ก.5.1 วิถีหาคุณภาพของการไหลเข้า (Inflow hydrograph) ที่นิยมใช้กันโดยทั่วไปคือเอกชลภาพ (Unit hydrograph) และในการศึกษาครั้งนี้จะหาคุณภาพของการไหลเข้าอ่างเก็บน้ำขนาดเล็กของ Komsartra(1969) โดยพบว่าแผนส่วนเกิน 1 มิลลิเมตรที่ตกบนพื้นที่รับน้ำจะทำให้เกิดน้ำท่าโดยความสัมพันธ์ระหว่าง Q/Q_p กับ t/t_p มีค่าเฉลี่ยคงที่ตามรูปที่ ก.4

ค่า t_p และ Q_p จะขึ้นกับลักษณะของลุ่มน้ำโดยมีความสัมพันธ์ตามสมการ ก.12 และ ก.13 คือ

$$t_p = 1.9 [L L_c / \sqrt{S}]^{0.162} \quad \dots\dots\dots \text{ก.12}$$

$$Q_p = 0.161 (A/t_p)^{0.98} \quad \dots\dots\dots \text{ก.13}$$

เมื่อ

t_p = เวลาตั้งแต่ฝนตกจนเกิดปริมาณน้ำนองสูงสุด, ชั่วโมง

Q_p = ปริมาณน้ำนองสูงสุด, ลบ.ม/วินาที/แผนส่วนเกิน 1 มม.

L = ความยาวตามลำน้ำจากจุดที่ตั้งโครงการจนถึงขอบเขตพื้นที่รับน้ำ

ทางต้นน้ำ, กิโลเมตร

L_c = ความยาวตามลำน้ำจากจุดที่ตั้งโครงการจนถึงจุดกึ่งกลางศูนย์กลางของพื้นที่รับน้ำ, กิโลเมตร

A = พื้นที่รับน้ำฝน, ตร.กม.

S = ความลาดชันเฉลี่ยของลำน้ำ

ชลภาพที่ได้จากวิธีนี้ถือว่าเกิดจากฝนส่วนเกิน (Excess rainfall) ที่กลายมาเป็นน้ำท่าเท่านั้น ความคลาดเคลื่อนในการหาชลภาพการไหลเข้าอ่างเก็บน้ำนี้ อยู่กับการประมาณฝนส่วนเกินซึ่งมีความยุ่งยากมากเนื่องจากขาดข้อมูลที่สมบูรณ์เพียงพอในการวิเคราะห์และยังขึ้นอยู่กับสภาพพื้นที่ลุ่มน้ำขนาดเล็กอีกด้วย ดังนั้นโดยทั่วไปเพื่อความปลอดภัยจึงมักสมมุติให้สัมประสิทธิ์น้ำท่าหรือฝนส่วนเกินที่กลายเป็นน้ำท่ามีค่า 80-90 % ของปริมาณฝนที่ตก (สัจจะ, 2524)

3.5.2 ปริมาณฝนที่ใช้ในการออกแบบ (Rainfall)

สำหรับปริมาณฝนที่ใช้ในการออกแบบนั้นโดยทั่วไปมักใช้ฝนที่ตกติดต่อกัน 2 ชม. ที่คาบการกลับ 25 ปี เนื่องจากอายุการใช้งานของอ่างเก็บน้ำไม่นานนักที่เลือกคาบการกลับที่มากกว่านี้จะทำให้ค่าก่อสร้างสูงและฝนที่มีช่วงเวลาการตก 2 ชม. นั้นจะทำให้อัตราการไหลสูงสุด (สัจจะ, 2524)

ดังนั้นปริมาณฝนที่ใช้ในการออกแบบจะหาได้จากสมการ

$$P = i \cdot d \quad \dots\dots\dots \text{ก. 14}$$

เมื่อ

P = ปริมาณฝนที่ตกทั้งหมด, มิลลิเมตร

i = ความหนักของฝน, มิลลิเมตรต่อ ชม.

d = ช่วงเวลาที่ฝนตก, ชม.

ปริมาณฝนส่วนเกินหาได้จากสมการที่อยู่ในรูป

$$R_g = C_r \cdot P \quad \dots\dots\dots \text{ก. 15}$$

เมื่อ

R_g = ปริมาณฝนส่วนเกินที่กลายเป็นน้ำท่า

C_r = สัมประสิทธิ์น้ำท่า

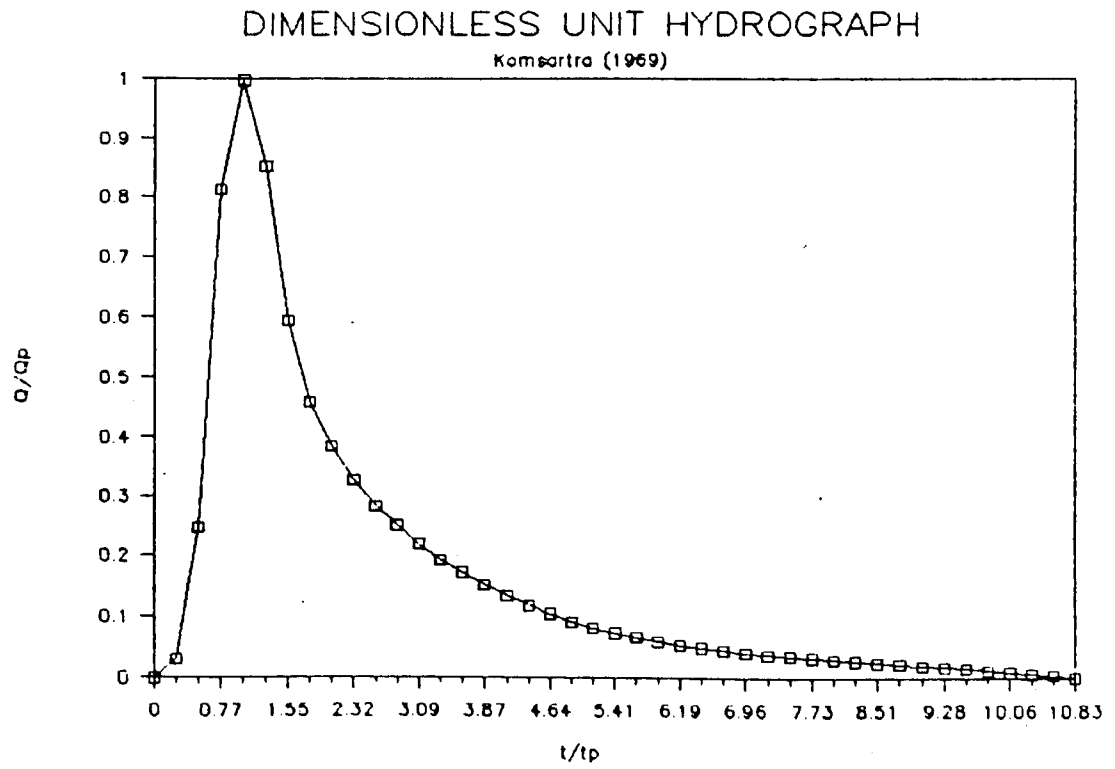
ชลภาพการไหลเข้าอ่างเก็บน้ำท่าซึ่งเกิดจากฝนที่ตกบนพื้นที่ลุ่มน้ำจะหาได้จากสมการ ก. 16

และ ก. 17

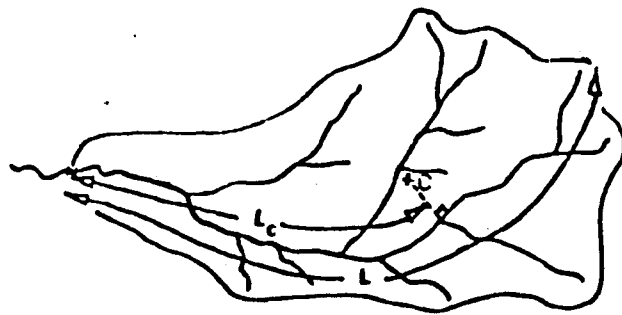
$$t = E \cdot t_p \quad \dots\dots\dots \text{ก. 16}$$

$$I = F \cdot R_g \cdot Q_p \quad \dots\dots\dots \text{ก. 17}$$

เมื่อ E และ F มีค่าเป็นค่า t/t_p และ Q/Q_p ตามรูปที่ ก.4 ส่วน I เป็นอัตราดิเนตของชลภาพของการไหลเข้าอ่างเก็บน้ำ



รูปที่ ก.4 ค่าเอกชลภาพไร้มิติของ Komsartra(1969)



รูปที่ ก.5 ลักษณะทางกายภาพของกลุ่มน้ำ

ก.5.4 อัตราการไหลของอ่างเก็บน้ำ (Reservoir routing) เป็นไปตามสมการสภาวะต่อเนื่อง (Continuity Equation) ซึ่งเป็นความสัมพันธ์ระหว่างกาไหลเข้า การไหลออกและการเก็บกักตามสมการ

$$I - O = \frac{ds}{dt} \quad \dots\dots\dots \text{ก.18}$$

เมื่อ I = อัตราการไหลเข้า ลบ.ม/วินาที
 O = อัตราการไหลออก ลบ.ม/วินาที

ds/dt = อัตราการเปลี่ยนแปลงปริมาณการเก็บกัก ลบ.ม/วินาที เพื่อความสะดวกในการคำนวณมักใช้ค่าเฉลี่ยของการไหลเข้าและการไหลออกในช่วงเวลา Δt ขึ้นๆ ซึ่งสามารถเขียนในรูปของ finite difference ได้เป็น

$$\frac{I_1 + I_2}{2} - \frac{O_1 + O_2}{2} = \frac{S_2 - S_1}{\Delta t} \quad \dots\dots\dots \text{ก.19}$$

สัญลักษณ์ 1 และ 2 แสดงเวลาเริ่มต้นและสิ้นสุดของช่วงการไหล Δt สมการ ก. 19 มีตัวแปรที่ไม่ทราบค่าคือ O_2 และ S_2 หรืออีกนัยหนึ่งก็คือ จะต้องหาปริมาณเก็บกัก S_2 ที่สัมพันธ์กับระดับน้ำในอ่างเก็บน้ำซึ่งจะหาได้โดยวิธีลองผิดและลองถูก

ก.5.5 อัตราการไหลผ่านทางระบายน้ำล้น กรณีระดับน้ำในอ่างเก็บน้ำสูงกว่าระดับสันทางระบายน้ำล้น จะเกิดการไหลล้นข้ามสันทางระบายน้ำล้น อัตราการไหลผ่านทางระบายน้ำล้นจะหาได้จาก

$$Q = C_1 \cdot BH^{3/2} \quad \dots\dots\dots \text{ก.20}$$

เมื่อ Q = อัตราการไหลผ่านทางระบายน้ำล้น ลบ.ม./วินาที
 C_1 = สัมประสิทธิ์การไหล (Discharge coefficient)
 B = ความกว้างทางระบายน้ำล้น ม.
 H = ความสูงของน้ำเหนือสันทางระบายน้ำล้น ม.

สมการที่ ก.20 เป็นสมการการไหลผ่านทางระบายน้ำล้นที่ไม่มีประตูบังคับ (Ungate Spillway) และอาจถือว่าค่า C_1 มีค่าคงที่ซึ่งมีค่าเท่ากับ 1.712 (SCS)

ก.6 การออกแบบทางระบายน้ำล้น

อาคารระบายน้ำล้น คือ อาคารที่สร้างสำหรับทำหน้าที่ระบายน้ำที่ไหลมามากจนอ่างเก็บน้ำรับไว้ไม่ได้ ทั้งลงไปยังลำน้ำด้านท้ายเขื่อนได้อย่างปลอดภัยเพื่อป้องกันไม่ให้น้ำในอ่างเก็บน้ำมีระดับสูงถึงสันเขื่อนซึ่งจะทำให้เกิดการไหลล้นข้ามสันเขื่อนและกัดเซาะเขื่อนให้เสียหาย อาคารระบายน้ำล้นประกอบด้วยทางน้ำไหลเข้า ลำราง ส่วนสลายพลังงาน

ก.6.1 ทางน้ำไหลเข้าแบบตรง (Straight Inlets) ทางน้ำไหลเข้าจะเป็นส่วน

เชื่อมต่อระหว่างอ่างเก็บน้ำกับลำรางและเป็นส่วนที่ควบคุมการไหล โดยจะออกแบบเป็นทางน้ำไหลเข้าแบบตรง (Straight Inlet) ของหน่วยงานอนุรักษ์ดินแห่งสหรัฐอเมริกา (SCS) ซึ่งสันทางระบายน้ำล้นมีลักษณะแบบราบ

จากรูปที่ ก.6 ปริมาณการไหลผ่านทางน้ำไหลเข้าแบบตรงไม่คิดระยะน้ำขึ้นน้ำลง

$$Q_{m1} = Q_{s1} (1.20 - 0.00984 Z) \dots\dots\dots \text{ก.21}$$

$$Q_{m1} = 1.712 W h^{3/2} \dots\dots\dots \text{ก.22}$$

$$q_{m1} = Q_{m1} / W \dots\dots\dots \text{ก.23}$$

เมื่อ Q_{m1} = ปริมาณการไหลผ่านทางน้ำไหลเข้าแบบตรงโดยไม่คิดระยะน้ำขึ้นน้ำลง (ลบ.ม./วินาที)

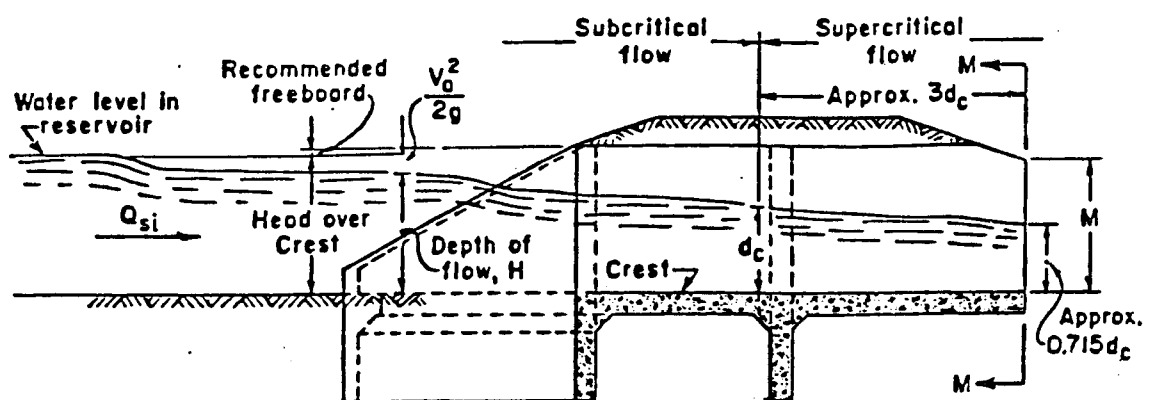
Q_{s1} = ปริมาณการไหลผ่านทางน้ำไหลเข้าแบบตรง (ลบ.ม./วินาที)

W = ความกว้างของสันทางระบายน้ำล้น (ม.)

h = ความสูงของน้ำเหนือระดับสันทางระบายน้ำล้น (ม.)

q_{m1} = ปริมาณการไหลผ่านทางน้ำไหลเข้าแบบตรงโดยไม่คิดระยะน้ำขึ้นน้ำลงต่อหนึ่งหน่วยความกว้าง (ลบ.ม./วินาที/ม.)

Z = ความแตกต่างระหว่างระดับสันทางระบายน้ำล้นกับระดับร่องน้ำธรรมชาติ (ม.)



รูปที่ ก.6 รูปตัดแสดงการไหลของทางน้ำไหลเข้าแบบตรง

ขนาดและ Dimension ต่างๆ ของทางน้ำไหลเข้าแบบตรงตามปริมาณการไหลผ่าน (q_{m1}) จากตารางที่ ก.4

ตารางที่ ก.4 ขนาดมิติปริมาณการไหลผ่านทางน้ำไหลเข้าแบบตรงต่อความกว้างหนึ่งหน่วย (SCS, CHUTE SPILLWAY SECTION 14)

h	M	q_{m1}
0.60	0.60	0.815
0.75	0.75	1.139
0.90	0.90	1.497
1.05	0.95	1.877
1.05	0.95	2.241
1.35	1.20	2.739
1.50	1.30	2.999

ก.6.2 วางเท (chute channel)

เป็นส่วนที่อยู่ต่อจากทางน้ำไหลเข้า (Inlet structure) โดยปกติความลาดชันของวางเทจะพิจารณาให้เหมาะสมกับความลาดเอียงของระดับพื้นดินเพื่อให้มีงานชดเชย แต่ความลาดของวางเทต้องเพียงพอที่จะควบคุมให้น้ำไหลด้วยความเร็วสูงกว่าความเร็ววิกฤติ

สำหรับการหาความลึกของการไหลในแต่ละส่วนของวางเทหา โดยวิธีสมดุลย์พลังงานดังแสดงในรูปที่ ก.7 (USBR "DESIGN OF SMALL DAM" 1974)

$$Z + d_1 + h_{v1} = d_2 + h_{v2} + \Delta h_L \dots \text{ก.24}$$

$$\Delta h_L = S \cdot L$$

$$h_{v1} = v_1^2 / 2g$$

$$h_{v2} = v_2^2 / 2g$$

$$S = (s_1 + s_2) / 2$$

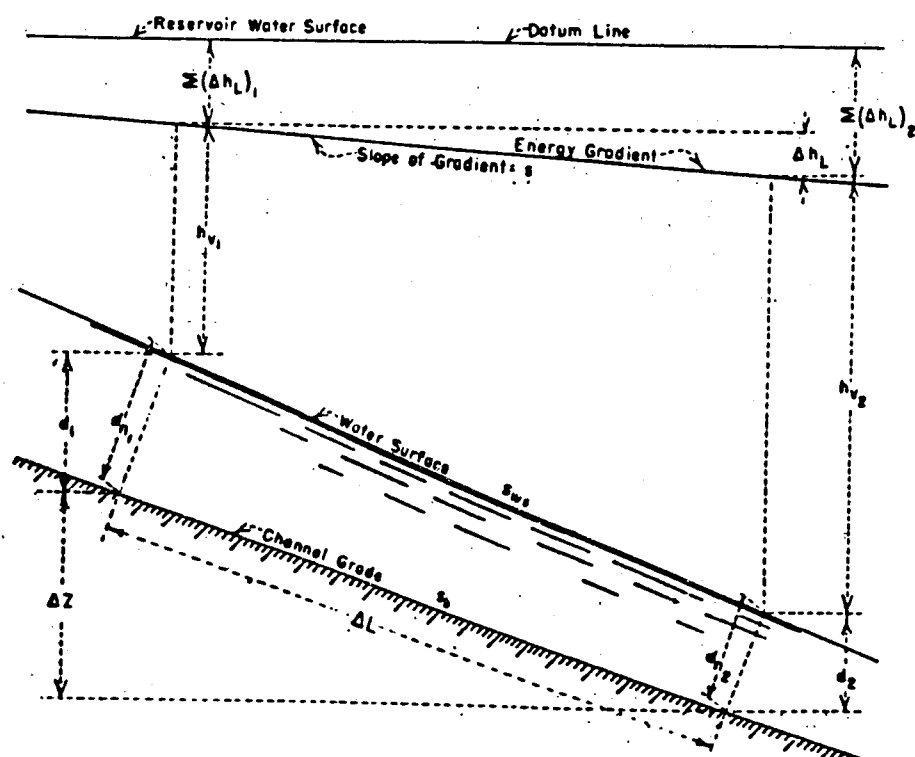
$$s_1 = [v_1 \cdot n / R_1^{2/3}]^2$$

$$s_2 = [v_2 \cdot n / R_2^{2/3}]^2$$

$$R_1 = A_1 / P_1$$

$$R_2 = A_2 / P_2$$

- เมื่อ Z = ค่าแตกต่างระหว่างจุดที่ 1 กับจุดที่ 2 ของช่วงรางเทที่มีพิจารณา
- d_1, d_2 = ความลึกของการไหลจุดที่ 1 และ 2 ตามลำดับ
- v_1, v_2 = ความเร็วของการไหลจุดที่ 1 และ 2 ตามลำดับ
- S = ค่าเฉลี่ยความลาดเสียตทาน
- n = สัมประสิทธิ์ความขรุขระ ของพื้นรางเท
- S_1, S_2 = ค่าความลาดเสียตทานจุดที่ 1 และ 2 ตามลำดับ
- R_1, R_2 = รัศมีเชิงกลศาสตร์จุดที่ 1 และ 2 ตามลำดับ
- A_1, A_2 = พื้นที่ของการไหลจุดที่ 1 และ 2 ตามลำดับ
- P_1, P_2 = เส้นขอบเปียกของการไหลจุดที่ 1 และ 2 ตามลำดับ



รูปที่ ก.7 การหาความลึกของการไหลในรางเทโดยวิธีสมดุลพลังงาน

ระยะโพ้น้ำของรางเท = $2.0 + 0.025 v \sqrt{d^3}$ ฟุต ก.25

- เมื่อ v = ความเร็วของการไหลที่จุดพิจารณา
- d = ความลึกของการไหลที่จุดพิจารณา

ก.6.3 รูปร่างรางระบายในสภาพการไหลในทางน้ำเปิด รูปร่างรางระบายมักจะเลือกให้เข้ากับสภาพธรรมชาติและทฤษฎีวิทยาโดยมากมักจะเป็นช่วงตรงต่อด้วยส่วนโค้งตั้ง (Vertical curve) ส่วนโค้งดังกล่าวจะเป็นโค้งเข้า (Concave) กับโค้งออก (Convex) การออกแบบส่วนโค้งนี้จะต้องไม่ให้ความโค้งเด่นหรือมากเกินไป เพราะจะก่อให้เกิดการแยกตัวจากผิวรางเท ผิวรางระบายส่วนโค้งออกที่เหมาะสมไม่ควรจะมากกว่าแนวการไหลของลำการไหลอิสระที่มีพลังงานจำเพาะเท่ากับการไหลผ่านจุดดังกล่าว (USBR "DESIGN OF SMALL DAM" 1974)

ก. ส่วนการออกแบบส่วนโค้งออก (Convex curve ประมาณได้ตามสมการ

$$-Y = x \tan \theta + \frac{x^2}{K(4E \cos^2 \theta)} \dots\dots\dots \text{ก.26}$$

เมื่อ θ = เป็นมุมของพื้นทางด้านเหนือน้ำของส่วนโค้ง

E = พลังงานจำเพาะของการไหลซึ่งเท่ากับ $d+h_v$

K = เป็นค่าสัมประสิทธิ์ซึ่งถ้าต้องการไม่ให้เกิดการแยกตัวควร จะมีค่ามากกว่าหรือเท่ากับ 1.5

ข. ส่วนการออกแบบส่วนโค้งเข้า (Concave curve) ประมาณได้ตามสมการ

$$R = 2 q \cdot v / p \dots\dots\dots \text{ก.27}$$

เมื่อ R = รัศมีส่วนโค้งเป็นฟุต

q = อัตราการไหลต่อฟุต

v = ความเร็วของการไหลเป็น ฟุต /วินาที

p = ความดันพลวัตในแนวตั้งฉากกับผิวน้ำเป็น ปอนด์/ฟุต² โดยทั่วไปเพื่อให้รัศมีความโค้งที่ดี ค่า p ที่เหมาะสมคือ 100 ปอนด์/ฟุต² ไม่ว่ากรณีใดส่วนโค้งเข้าของทางระบายน้ำไม่ควรจะมีค่าน้อยกว่า 10 เท่าของความลึกของการไหล

ค. ส่วนลู่และขยายออกรางระบาย (Convergence and Divergence)

สำหรับการลู่เข้าหรือขยายออกของรางระบายน้ำอัตราการบีบเข้า หรือการขยายออกจะต้องไม่เร็วเกินไปจนก่อให้เกิดการเปลี่ยนสภาพการไหล ตัวกำหนดที่ใช้คือค่า Froude number ($IF = v / \sqrt{gd}$) โดยมุมของการลู่เข้าหรือการขยายออกจะต้องมีค่าไม่เกินมุมตามความสัมพันธ์ดังนี้

$$\tan \theta = \frac{1}{3IF} \dots\dots\dots \text{ก.28}$$

ก.6.4 ส่วนสลายพลังงาน (Stilling basin)

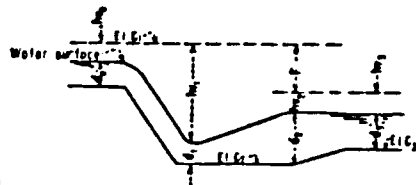
ส่วนสุดท้ายของทางระบายน้ำล้นคือส่วนสลายพลังงาน ในอ่างเก็บน้ำขนาดเล็กส่วนสลายพลังงานนั้นจะอาศัยการเกิด hydraulic jump แบบบังคับในการสลายพลังงานเพื่อให้ความเร็วของการไหลที่ออกไปต่ำพอที่จะไม่ก่อให้เกิดความเสียหายกับอาคารด้านท้ายน้ำซึ่งในการศึกษาครั้งนี้ใช้รูปแบบของ USBR (USBR "DESIGN OF SMALL DAM" 1974)

สมการ hydraulic jump มีดังนี้คือ

$$\begin{aligned} d_2/d_1 &= \frac{1}{2} \left[\sqrt{1+8IF_1^2} - 1 \right] \dots\dots\dots \text{ก.29} \\ \text{เมื่อ } d_2 &= \text{ความลึกของน้ำหลัง jump ตามรูปที่ ก.10} \\ d_1 &= \text{ความลึกของน้ำก่อน jump ตามรูปที่ ก.10} \\ IF_1 &= \text{Froude number ของการไหลก่อนเกิด jump} \end{aligned}$$

Relation of energy loss, critical depth, and depth before and after jump, for hydraulic jumps in rectangular channels with level floor.

E = Difference in energy levels of upstream and downstream ends of jump.
 d_c = Critical depth for flow considered. Based on pool width $d_c = \sqrt[3]{\frac{Q^2}{g}}$
 d_1 = Depth of upstream end of jump
 d_2 = Depth of downstream end of jump
 $K = d_2 - d_1$
 $E = E_1 - E_2 = (d_1 + h_{f1}) - (d_2 + h_{f2} + F)$
 $\frac{d_2}{d_1} = \sqrt{\frac{K}{K+1}}$
 $\frac{F}{K} = \frac{(K-1)^2}{4K} \sqrt{\frac{K}{K+1}}$



Data from memorandum by C.R. Burby, April 6, 1939

รูปที่ ก.8 พลังงานที่สูญเสียไปเนื่องจากการเกิด hydraulic jump

(USBR, "DESIGN OF SMALL CANAL STRUCTURE" 1978)

$$IF_1 = (v_1 / \sqrt{gd_1}) \dots\dots\dots \text{ก.30}$$

เมื่อ v_1 = ความเร็วของน้ำก่อนเกิด jump

d_1 = ความลึกของน้ำก่อนเกิด jump

g = ความเร่งของโลก 9.807 เมตร/วินาที²

จากสมการที่ ก.29 และสมการที่ ก.30 รูปแบบของ jump เปลี่ยนแปลงตามค่า IF_1 การคำนวณออกแบบส่วนสลายพลังงานแบบขั้นระนาบ จะใช้อ่างน้ำนิ่งมาตรฐานของ USBR ซึ่งกำหนดขนาดเหมาะสมกับ jump ที่เกิดจาก IF_1 ค่าต่างๆ ดังต่อไปนี้

ก. USBR II ตามรูปที่ ก.9 ซึ่งใช้สำหรับค่า IF_1 มากกว่า 4.5

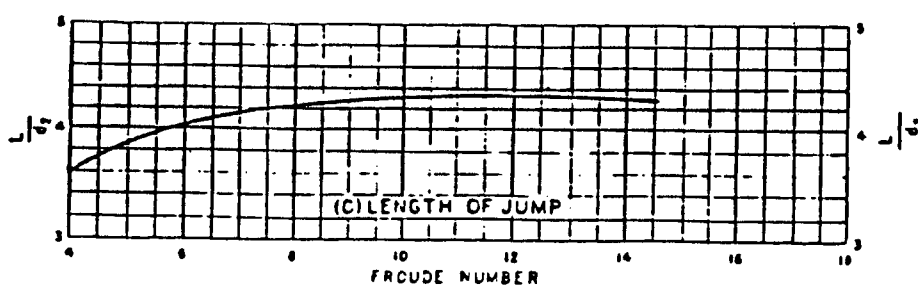
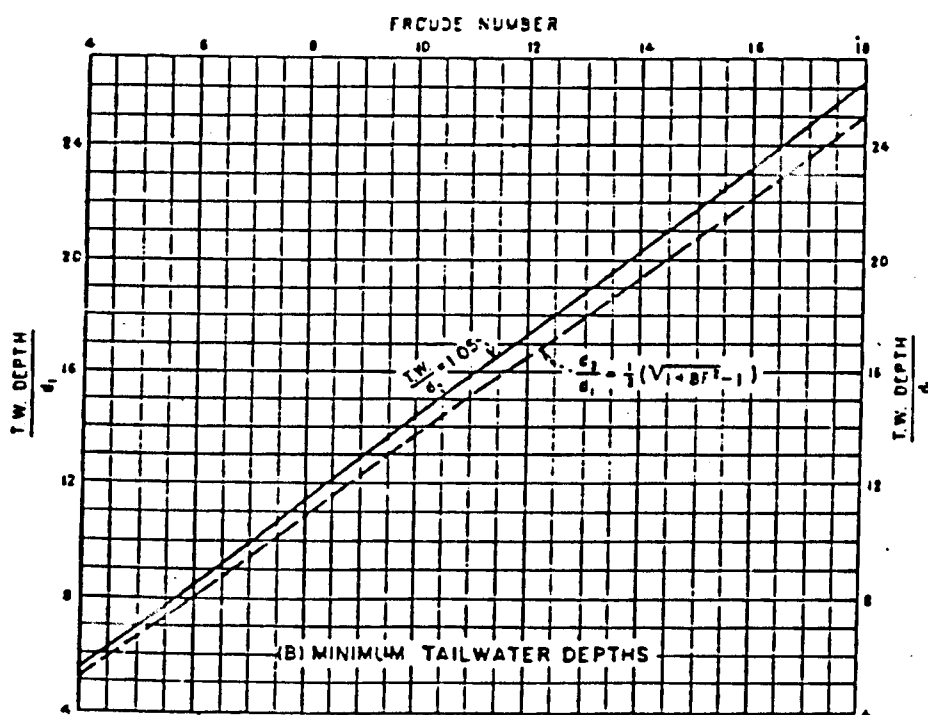
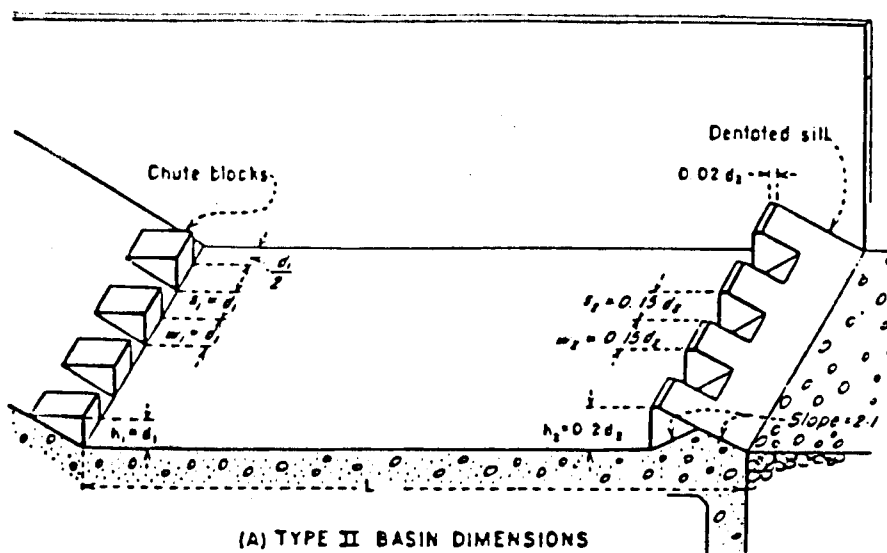
ข. USBR III ตามรูปที่ ก.10 ซึ่งใช้สำหรับค่า IF_1 มากกว่า 4.5 และความเร็วของการไหลก่อน jump ไม่เกิน 15-18 เมตร/วินาที

ค. USBR IV ตามรูปที่ ก.11 ซึ่งใช้สำหรับค่า IF_1 เท่ากับ 2.5 ถึง 4.5

ระยะพักน้ำของอ่างน้ำนิ่ง = 0.10 $(v_1 + d_2)$ ก.31

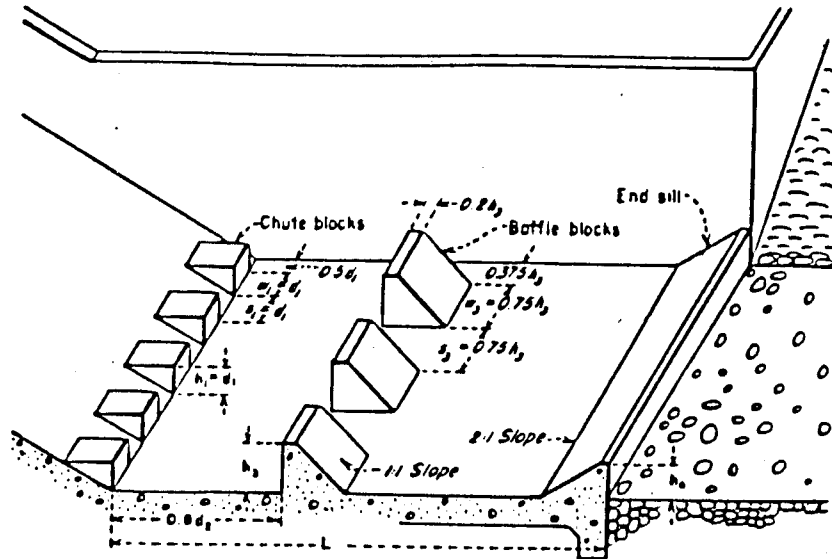
เมื่อ v_1 = ความเร็วของน้ำก่อนเกิด jump เมตร/วินาที

d_2 = ความลึกของน้ำหลังเกิด jump เมตร

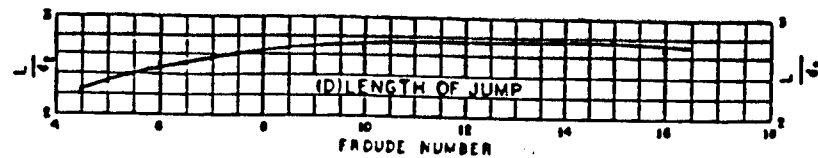
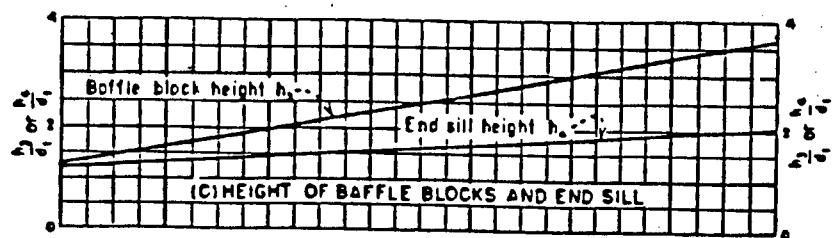
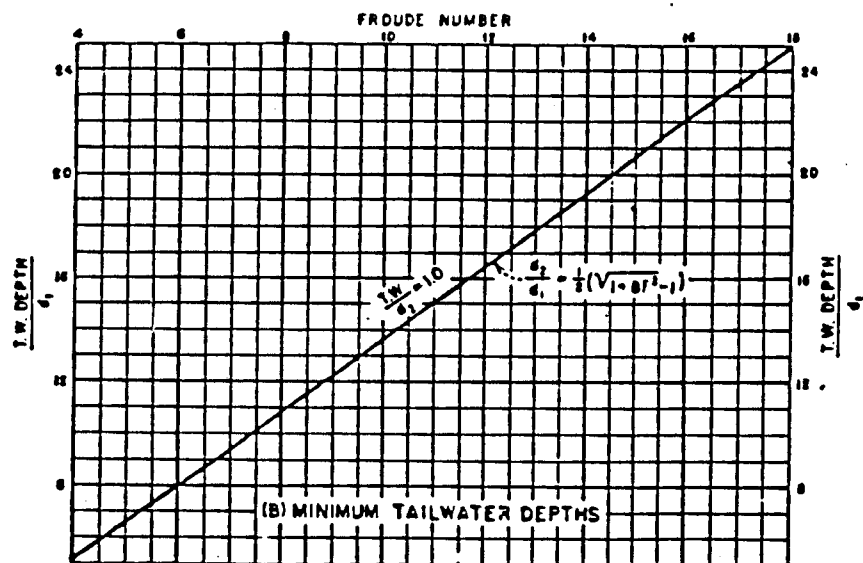


Stilling basin characteristics for use with Froude numbers above 4.5

รูปที่ ก.9 อ่างน้ำนิ่งแบบที่ II (USBR "DESIGN OF SMALL DAM" 1974)

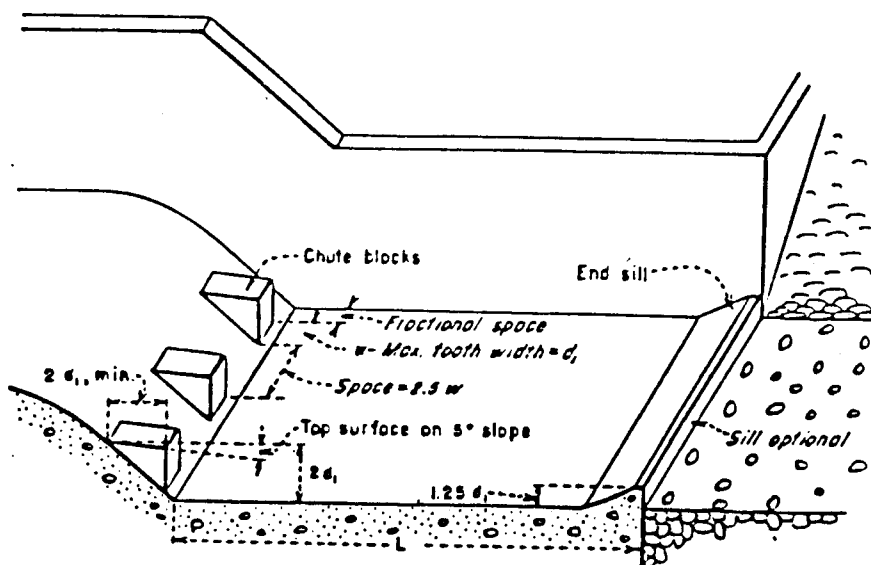


(A) TYPE III BASIN DIMENSIONS

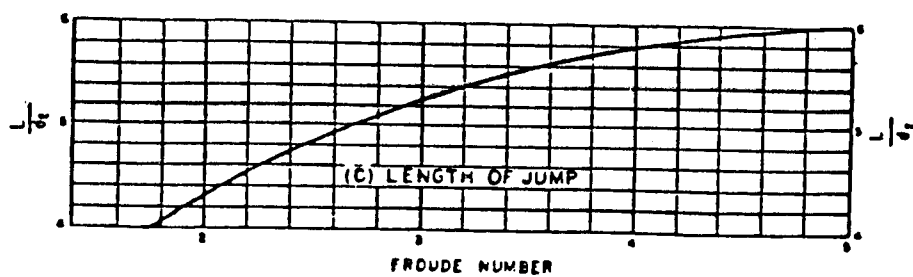
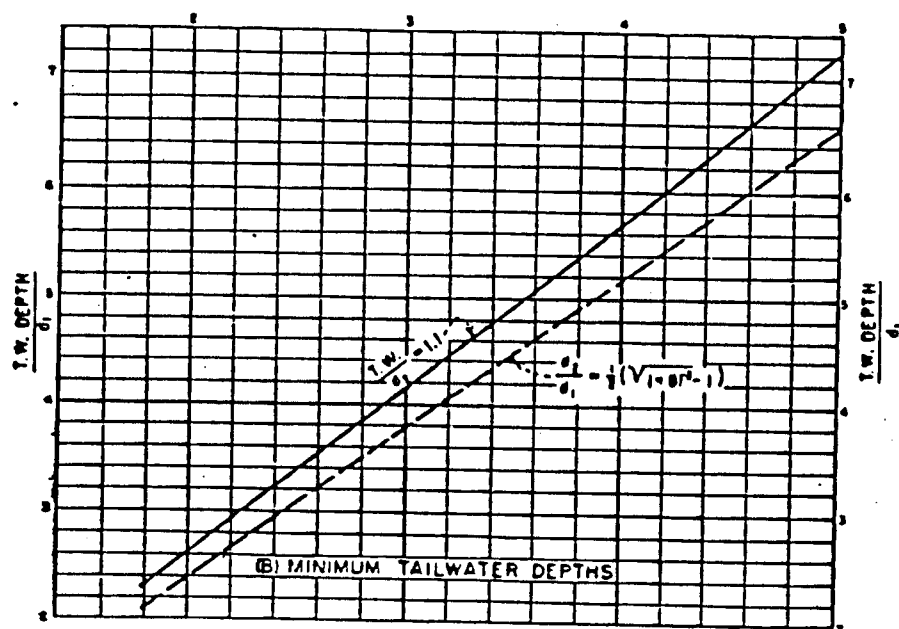


Stilling basin characteristics for use with Froude numbers above 4.5.

รูปที่ ก.10 อ่างน้ำนิ่งแบบที่ III (USBR "DESIGN OF SMALL DAM "1974



(A) TYPE IV BASIN DIMENSIONS
FROUDE NUMBER



รูปที่ ก.11 อ่างน้ำแบบที่ IV (USBR " DESIGN OF SMALL DAM" 1974)

ก.6.5 ตรวจสอบแรงดันยก (Uplift pressure) ทางน้ำซึมใต้อาคารทางระบายน้ำล้น (Percolation path) ควรยาวพอไม่ทำให้น้ำซึมแรงจนจะเอาเม็ดดินฐานรากให้ไหลหลุดออกไปกับน้ำซึมนอกจากนี้ความยาวนี้ยังควบคุมปริมาณน้ำที่สูญเสียจากการรั่วซึมได้ทั้งทางระบายน้ำล้นและควบคุมแรงดันขึ้นของน้ำใต้ดินอาคาร การกำหนดความหนาและความยาวของทั้งทางระบายน้ำล้นเพื่อต้านแรงดันของน้ำใต้ดินทางระบายน้ำล้นและป้องกันการพัฒนาเอาเม็ดดินฐานรากออกไปตามวิธีการของ Lane ดังนี้

$$\text{Weighted creep ratios} = \frac{\text{ระยะทางของน้ำซึมผ่านทั้งหมด}}{\text{เสียดหน้าโครงสร้าง}} \dots \text{ก.32}$$

หลักการหาระยะทางของน้ำซึมผ่าน

ก.1 คุณระยะตามแนวติดต่อระหว่างฐานรากกับอาคารแนวตั้งหรือแนวที่ทำมุมกับแนวราบมากกว่า 45 องศา

ข. 1/3 คุณระยะซึมตามแนวติดต่อระหว่างดินฐานรากกับอาคารแนวนอนหรือแนวที่ทำมุมกับแนวราบน้อยกว่า 45 องศา

จากการศึกษาค่าที่ใช้ในการออกแบบสำหรับอัตราส่วน Weighted creep Ratio ของวัสดุต่างๆ ใช้ค่าตามตารางที่ ก.5

ตารางที่ ก.5 อัตราส่วน Weighted creep Ratio สำหรับวัสดุต่าง ๆ

วัสดุ	อัตราส่วน Weighted creep Ratio
ทรายละเอียดมากหรือ silt	8.5
ทรายละเอียด	7.0
ทรายปานกลาง	6.0
ทรายหยาบ	5.0
กรวดละเอียด	4.0
กรวดปานกลาง	3.5
กรวดหยาบรวมถึงหินก้อนเล็ก	3.0
หินก้อน โดมมีหินก้อนเล็กและกรวดปน	2.5
ดินเหนียวอ่อน	3.0
ดินเหนียวปานกลาง	2.0
ดินเหนียวแข็ง	1.8
ดินเหนียวแข็งมากหรือดินดาล	1.6

ภาคผนวก ข แบบจำลองน้ำท่ารายวัน

ข.1 การพัฒนาแบบจำลองน้ำท่า

แบบจำลองน้ำท่ารายวัน ของนาถฤทธิ (2529) ได้พัฒนามาจากแบบจำลองของ SCS ซึ่งเป็นแบบจำลองที่หาปริมาณน้ำท่า โดยขึ้นอยู่กับลักษณะของดินและการใช้ประโยชน์ที่ดินรวมทั้งเงื่อนไขความชื้นก่อนหน้าด้วย

แผนที่ตกลงมาจะแยกออกเป็น 3 ส่วนคือ การสูญเสียเริ่มแรก (Initial abstraction) การซึมลงดิน (Infiltration) และน้ำท่า (runoff) ดังนั้นจากการสมมติปริมาณน้ำ จะได้นี้

$$P = I_a + F + Q \dots\dots\dots \text{ข.1}$$

เมื่อเกิดน้ำท่าแล้วค่าปริมาณการซึมสะสมเพิ่มขึ้นจนถึงจุดหนึ่งที่เรียกว่า ศักยภาพซึมสูงสุด และปริมาณน้ำท่าสะสมจะเพิ่มขึ้นเมื่อปริมาณฝนสะสมเพิ่มขึ้น SCS ได้ตั้งสมมติฐานของความสัมพันธ์ระหว่างตัวแปรดังกล่าวดังต่อไปนี้

$$\frac{F}{S} = \frac{Q}{P - I_a} \dots\dots\dots \text{ข.2}$$

โดยที่ P = เป็นปริมาณน้ำฝนทั้งหมด (total rainfall) มม.

I_a = เป็นการสูญเสียเริ่มแรก (Initial abstraction) ซึ่งจะรวมการดัก (interception) การเก็บกักในที่ต่ำ (depression storage) และซึมก่อนจะมีน้ำท่าเกิดขึ้น , มม.

F = เป็นปริมาณการซึมสะสมที่ไม่รวม I_a , มม.

Q = เป็นปริมาณน้ำท่า มม.

S = เป็นศักยภาพการเก็บกักหรือการซึมสูงสุด (Potential moisture retention or Infiltration)

จากสมการทั้งสองสมการหาค่า Q และ F ได้ดังนี้

$$Q = 0 \quad \text{ในกรณี} \quad P < I_a \dots\dots\dots \text{ข.3}$$

$$Q = \frac{(P - I_a)^2}{P - I_a + S} \quad \text{ในกรณี} \quad P > I_a \dots\dots\dots \text{ข.4}$$

$$F = 0 \quad \text{ในกรณี} \quad P < I_a \dots\dots\dots \text{ข.5}$$

$$F = \frac{(P - I_a)S}{P - I_a + S} \quad \text{ในกรณี} \quad P > I_a \dots\dots\dots \text{ข.6}$$

SCS แนะนำให้ใช้ความสัมพันธ์ระหว่าง I_a กับ S ดังนี้

$$I_a = 0.2 S \dots\dots\dots \text{ข.7}$$

ค่าพารามิเตอร์ S หาได้จากความสัมพันธ์ ดังนี้

$$S = \frac{25400}{CN} - 254 \text{ เป็นมิลลิเมตร} \dots\dots\dots \text{ข.8}$$

CN

โดยที่ CN เป็นพารามิเตอร์ที่เรียกว่า CURVE NUMBER ซึ่งขึ้นอยู่กับชนิดของดินและการใช้พื้นที่ รวมทั้งเงื่อนไขของความชื้นในดินเริ่มแรก (antecedent soil moisture)

ข.2 การประมาณค่า CURVE NUMBER (CN)

ปริมาณน้ำท่าขึ้นอยู่กับชนิดของดิน (soil type) การใช้พื้นที่ (land use) และเงื่อนไขความชื้นในดินเริ่มแรก ดังนั้นการหาปริมาณน้ำท่าจึงต้องมีดัชนีบอกลักษณะทั้งสามข้างต้นซึ่งในที่นี้ก็คือ CURVE NUMBER (CN) นั้นเอง

ข.2.1 การจำแนกชนิดของดิน (Soil type classification)

สำหรับแบบจำลองน้ำท่านี้ได้จำแนกชนิดของดินออกเป็น 4 กลุ่มใหญ่ ๆ ดังต่อไปนี้

กลุ่ม A เป็นดินที่น้ำซึมผ่านได้สูง เช่น กรวด (gravel) ทรายเม็ดปน (silt) เป็นต้น

กลุ่ม B เป็นดินที่ยอมให้น้ำซึมผ่านได้ปานกลาง เช่น ดินร่วนปนทราย เป็นต้น

กลุ่ม C เป็นดินที่ยอมให้น้ำซึมผ่านได้ค่อนข้างต่ำ เช่น ดินร่วนปนดินเหนียว เป็นต้น

กลุ่ม D เป็นดินที่ยอมให้น้ำซึมผ่านได้ต่ำ เช่น ดินเหนียว เป็นต้น

ข.2.2 การจำแนกการใช้พื้นที่ (Land use classification)

แบบจำลองน้ำท่านี้ได้จำแนกการใช้พื้นที่ออกเป็น 4 กลุ่มดังนี้

1. พื้นที่ป่า (Wood / forest)
2. พื้นที่เกษตรกรรม (Agriculture)
3. พื้นที่โล่งเตียนและที่อยู่อาศัย (bare land and residential)
4. พื้นที่เป็นน้ำ (water surface)

ข.2.3 เงื่อนไขความชื้นก่อนหน้า (antecedent moisture content, AMC) SCS ได้แบ่งเงื่อนไขความชื้นก่อนหน้าออกเป็น 3 กรณี โดยพิจารณาปริมาณฝนก่อนหน้า 5 วัน (5- day antecedent rainfall) ดังแสดงในตารางที่ ข.1

ตาราง ข.1 การจำแนกเงื่อนไขความชื้นก่อนหน้า

AMC	ปริมาณฝนก่อนหน้า 5 วัน (มม.)
I	น้อยกว่า 12
II	อยู่ระหว่าง 12 กับ 38
III	มากกว่า 38

ข.2.4 ค่า CURVE NUMBER ที่ใช้ในการหาปริมาณน้ำท่า

ค่า CN สำหรับเงื่อนไขที่ II (CN_{II}) สามารถหาได้จากตารางที่ ข.2

ตารางที่ ข.2 การใช้พื้นที่และชนิดของดิน

การใช้พื้นที่ (land use)	ชนิดของดิน (Soil type)			
	A	B	C	D
พื้นที่เป็นป่า	25	55	70	77
พื้นที่เกษตรกรรม	64	75	82	86
พื้นที่โล่งเตียน, ที่อยู่อาศัย	75	84	88	91
พื้นที่เป็นน้ำ	98	98	98	98

สำหรับค่า CURVE NUMBER (CN) เงื่อนไข I (CN_I) และ III (CN_{III}) Richard H, Hawkins Allin T. Hjelmfelt, and Adrain W Zevenbergen (1985) ได้ให้ความสัมพันธ์กับ CN_{II} ดังนี้

$$CN_I = \frac{CN_{II}}{2.281 - 0.0128 * CN_{II}} \quad \dots\dots\dots \text{ข.9}$$

$$CN_{III} = \frac{CN_{II}}{0.427 + 0.00573 * CN_{II}} \quad \dots\dots\dots \text{ข.10}$$

ข.3 การไหลของน้ำผิวดิน (Surface runoff routing)

การไหลของน้ำผิวดิน หาคความสัมพันธ์ได้จากวิธีการของ Muskingum ดังนี้

$$QQ(j,k) = QQ(j,k-1) + \frac{I_m - QQ(j,k-1)}{K + 0.5} \dots\dots\dots \text{ข.11}$$

$$I_m = (QQ(j,k) + QQ(j,k-1))/2 \dots\dots\dots \text{ข.12}$$

โดยที่

$QQ(j,k)$ = เป็นการไหลผิวดินในเดือนที่ j วันที่ k

$QQ(j,k-1)$ = เป็นการไหลผิวดินในเดือนที่ j วันที่ $k-1$

K = เป็นเวลาของการเก็บกัก (Time of Storage) ซึ่งจะได้จากสมการของ SCS ดังนี้

$$K = (1/27360) * (3281 * L)^{0.8} * (S+1)^{0.7} / S_0^{0.5} \dots\dots\dots \text{ข.13}$$

โดยที่

K = เวลาเก็บกัก วัน

L = ความยาวของแม่น้ำหลัก กิโลเมตร

S = ศักย์เก็บกักสูงสุด มม.

S_0 = เป็นความชันของพื้นที่ลุ่มน้ำ เปอร์เซนต์

ข.4 การไหลผิวดิน (surface runoff)

เมื่อปริมาณฝน (P) มากกว่า การสูญเสียเริ่มแรก (I_u) จะเกิดการไหลผิวดินขึ้นซึ่งสามารถหาได้ดังนี้

$$QQ(j,k) = Q(j,k) - Q(j,k-1) \dots\dots\dots \text{ข.14}$$

โดยที่

$QQ(j,k)$ = เป็นการไหลผิวดิน ในเดือนที่ j วันที่ k

$Q(j,k)$ = เป็นการไหลผิวดินสะสมในเดือนที่ j วันที่ k

$Q(j,k-1)$ = เป็นการไหลผิวดินสะสมในเดือนที่ j วันที่ $k-1$

ข.5 การซึมในส่วนเก็บกักในดิน (Infiltration)

ปริมาณการซึมสามารถหาได้จากความสัมพันธ์ดังนี้

$$FF(j,k) = F(j,k) - F(j,k-1) \dots\dots\dots \text{ข.15}$$

โดยที่

$FF(j,k)$ = เป็นปริมาณการซึมในเดือนที่ j วันที่ k

$F(j,k)$ = เป็นปริมาณการซึมสะสมในเดือนที่ j วันที่ k

$F(j,k-1)$ = เป็นปริมาณการซึมสะสมในเดือนที่ j วันที่ $k-1$

ข.6 ส่วนเก็บกักในดิน (Soil moisture storage)

ระดับน้ำในส่วนเก็บกักในดิน จะเปลี่ยนแปลงในแต่ละวันดังนี้

$$HSM(j,k) = HSM(j,k-1) + FF(j,k) - EVP(j,k) - REC(j,k) \dots \text{ข.16}$$

โดยที่

$HSM(j,k)$ = เป็นระดับน้ำในส่วนเก็บกักในเดือนที่ j วันที่ k

$HSM(j,k-1)$ = เป็นระดับน้ำในส่วนเก็บกักในเดือนที่ j วันที่ $k-1$

$FF(j,k)$ = ปริมาณการซึม ในเดือนที่ j วันที่ k

$EVP(j,k)$ = การคายระเหย ในเดือนที่ j วันที่ k

$REC(j,k)$ = เป็นการซึมจากส่วนเก็บกักในดินลงสู่ส่วนเก็บกักน้ำใต้ดินในเดือนที่ j วันที่ k

ข.7 การคายระเหย (Evapotranspiration)

การคายระเหยในแต่ละวัน หาได้ดังนี้

$$EVP(j,k) = Kp * E_{pan}(j,k) \dots \text{ข. 17}$$

โดยที่

$EVP(j,k)$ = เป็นการระเหย ในเดือนที่ j วันที่ k

$E_{pan}(j,k)$ = เป็นค่าการระเหยจากสภาพ class A ในเดือนที่ j วันที่ k

Kp = เป็นค่าสัมประสิทธิ์ของพืชอ้างอิง

ข.8 การซึมส่วนเก็บกักในดินลงสู่ส่วนเก็บกักน้ำใต้ดิน

$$REC(j,k) = K_1 * XX * (HSM(j,k) - SFC) \dots \text{ข.18}$$

โดยที่

$REC(j,k)$ = เป็นปริมาณการซึมจากส่วนเก็บกักในดินสู่ส่วนเก็บกักน้ำใต้ดิน
ในเดือนที่ j วันที่ k

$XX = HSM(j,k) / SSAT$

SFC = เป็นระดับน้ำน้อยที่สุด ของส่วนเก็บกักในดิน ซึ่งกำหนดให้เท่ากับ $0.1 * SSAT$

K_1 = เป็นค่าสัมประสิทธิ์การซึมจากส่วนเก็บกักในดินลงสู่ส่วนเก็บกักน้ำใต้ดิน

ข.9 การไหลจากส่วนเก็บกักใต้ดิน (Groundwater storage routing)

$$G(j,k) = K_2 * HG(j,k) * e^{-(HSM(j,k)/SSAT)} \dots \text{ข.19}$$

โดยที่

$G(j,k)$ = เป็นปริมาณน้ำจากส่วนเก็บกักใต้ดิน ในเดือนที่ j วันที่ k

$HG(j,k)$ = เป็นส่วนเก็บกักของน้ำใต้ดินในเดือนที่ j วันที่ k

K_z = เป็นค่าสัมประสิทธิ์การหลากของส่วนเก็บกักน้ำใต้ดิน

ข.10 ปริมาณน้ำท่าทั้งหมด (total runoff)

ปริมาณน้ำท่าทั้งหมดคือผลบวกของปริมาณการไหลผิวดินกับปริมาณน้ำใต้ดิน

$$OU(j,k) = QQ(j,k) + G(j,k) \dots\dots\dots \text{ข.20}$$

โดยที่

$OU(j,k)$ = เป็นปริมาณน้ำท่าทั้งหมดในเดือนที่ j วันที่ k มีหน่วยเป็นความ

ลึกเพราะจะได้ปริมาณน้ำท่าหน่วยเป็นลูกบาศก์เมตรต่อวัน

$$OU(j,k) = OU(j,k) * AREA * CA * 1000 \dots\dots \text{ข.21}$$

โดยที่

$AREA$ = เป็นพื้นที่ลุ่มน้ำ , ตารางกิโลเมตร

CA = เป็นสัมประสิทธิ์พื้นที่ (Contributive area coefficient)

ซึ่งค่า CA นี้จะได้จากการเปรียบเทียบค่าจากการสังเกตข้อมูลน้ำท่า เนื่องจากในการศึกษารังนี้ไม่มีข้อมูลน้ำท่าจึงใช้ค่า CA ของภาคพื้น

$OU(j,k)$ = เป็นปริมาณน้ำท่า (ลูกบาศก์เมตรต่อวัน)

จากนั้นคำนวณหาปริมาณน้ำท่าสะสมรายวันดังนี้

$$CUM = CUM + OU(j,k) \dots\dots\dots \text{ข.22}$$

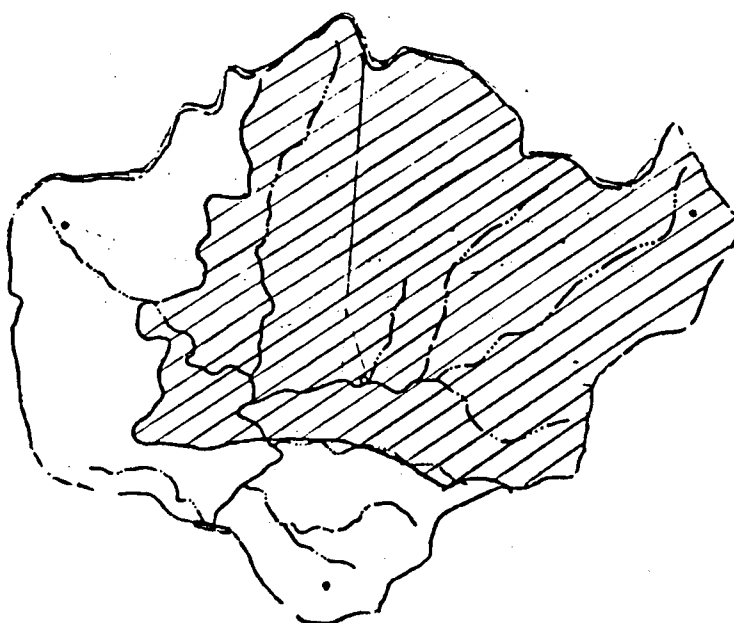
โดยที่

CUM = เป็นปริมาณน้ำท่าสะสมรายวัน

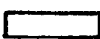
ภาคผนวก ค.

แผนที่การใช้ที่ดิน (Land use) และชนิดของดิน (Soil type) ของลุ่มน้ำห้วย
แม่ผาแพน

- รูปที่ ค.1 การใช้ที่ดินลุ่มน้ำห้วยแม่ผาแพน
- รูปที่ ค.2 ชนิดของดินลุ่มน้ำห้วยแม่ผาแพน
- รูปที่ ค.3 ลักษณะทางกายภาพของลุ่มน้ำห้วยแม่ผาแพน
- รูปที่ ค.4 เส้นชั้นความสูงบริเวณเก็บกักของอ่างเก็บน้ำห้วยแม่ผาแพน
- รูปที่ ค.5 ผลการทดสอบบ่อน้ำดื่มที่ใช้ทำตัวเชื่อม และฐานราก
- รูปที่ ค.6 ความถี่น้ำ-ช่วงเวลา-ความถี่ของฝน ของจังหวัดเชียงใหม่



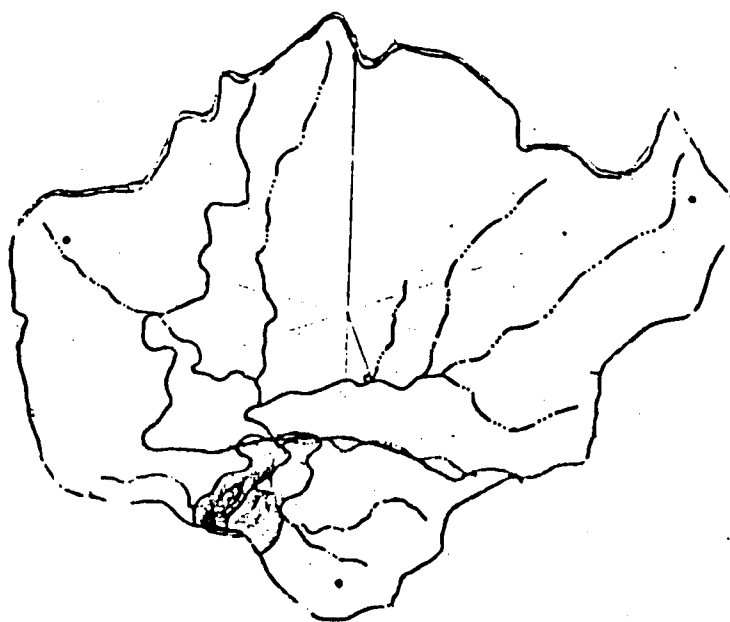
 พื้นที่ป่ามากกว่า 75 %

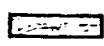
 พื้นที่ป่า 25 % ถึง 75 %

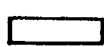
รูปที่ ค.1 การใช้ที่ดินลุ่มน้ำท้ายแม่ผาแพน

หมายเลข 50 = ดินชุดท้ายาง, น้ำซึมผ่านได้ค่อนข้างต่ำ = C

หมายเลข 60 = ดิน Granite derived soils complex, น้ำซึมผ่านได้ปานกลาง = B



 ดินหมายเลข 60

 ดินหมายเลข 50

จากรูปที่ ค.1 และรูปที่ ค.2 $WB = 22.04$ กม.²

$WC = 0.84$ กม.²

รูปที่ ค.2 ชนิดของดินลุ่มน้ำห้วยแม่ผาเทน



L = ความยาวลำน้ำสายหลัก = 6.9 กม.

L_c = ความยาวจากจุดทางออกจนถึงจุดที่เส้นตั้งฉากจุดศูนย์มวลถึงลำน้ำหลัก = 2.93 กม.

S = ความชันของลำน้ำหลัก = 6.08 %

พื้นที่รับน้ำ = 22.88 กม.²

รูปที่ ค.3 ลักษณะทางกายภาพของลุ่มน้ำห้วยแม่ผาแดง



รูปที่ ๙.๔ แสดงเส้นความสูงบนรีเลย์เก็บกักของอ่างเก็บน้ำห้วยแม่ปาน

```

      IF(PRINT.EQ.1)WRITE(12,2101)I
      WRITE(*,2101)I
2101  FORMAT(/,T5,'XXX LINE',I3,'NOT INTERSECTED BY TRIAL CIRCLE')
      8  CONTINUE
      DO 400 I=1,NOL
      IF(LNU(I).EQ.0)GO TO 400
      R1=SQRT((CX-C(I,3))**2+(CY-C(I,4))**2)
      R2=SQRT((CX-C(I,5))**2+(CY-C(I,6))**2)
      IF(R.LT.R1.AND.R.LT.R2)GO TO 400
      LNU(I)=0
      IF(SLOPE(I).EQ.BIGNO)LNU(I)=I
      IF(PRINT.EQ.1)GO TO 3006
      IF(SLOPE(I).EQ.BIGNO)WRITE(*,403)LNU(I)
      GO TO 3007
3006  IF(SLOPE(I).EQ.BIGNO)WRITE(12,403)LNU(I)
      403  FORMAT(/,'**** LINE',I3,'IS IN ARC BUT VERT. AND NOT USED')
3007  WRITE(12,401)LNU(I)
      401  FORMAT(/,T5,'*** LINE',I3,'IS NOT INTERSECTED BUT IS IN ARC')
      400  CONTINUE
C      FIND WIDTH OF SECTIONS BETWEEN LINE INTERSECTIONS SO SLICE ARRAY
C          CAN BE SET UP.
C      FIND LOCATION OF ARC EXIT AND NEAREST ADJACENT INTERSECTION.
C      NOTE K1 = TOTAL ENTRIES IN CIRCLE-LINE INTERSECTION ARRAY(ARCINT)
C      SORT ARC INTERSECTIONS FROM LEAST TO GREATEST X-COORDINATES
      K1M=K1-1
      24  DO 26 KY=1,K1M
      IF(ARCINT(KY,2).LE.ARCINT(KY+1,2))GO TO 26
      DO 25 KX=1,3
      SAVE=ARCINT(KY,KX)
      ARCINT(KY,KX)=ARCINT(KY+1,KX)
      ARCINT(KY+1,KX)=SAVE
      25  CONTINUE
      GO TO 24
      26  CONTINUE
      IF(PRINT.EQ.1)WRITE(12,2112)
      WRITE(*,2112)
2112  FORMAT(/,T5,'ARC INTERSECT WITH LINE ARRAY',/,T4,'LINE NO',T19,
      A'X',T32,'Y')
      IF(PRINT.EQ.1)WRITE(12,2114)((ARCINT(KZ,JJ),JJ=1,3),KZ=1,K1)
      WRITE(*,2114)((ARCINT(KZ,JJ),JJ=1,3),KZ=1,K1)
2114  FORMAT(T5,F3.0,T13,F10.3,2X,F10.2)
      LINE1=ARCINT(1,1)
      S1=ARCINT(1,2)
      S2=ARCINT(1,3)
      IF(PRINT.EQ.1)WRITE(12,8053)
      WRITE(*,8053)
      8053  FORMAT(/,T5,'THE ARRAY WITH ALL INTERSECTIONS FOLLOWS:')
      ICOUN=0
      K=1
      KK=0
      LL=NLIT+K1
C      COMBINE ARCINT AND INTAR IN ORDER OF INCREASING X-VALUES
      DO 70 I=1,LL
      KK=KK+1
      DO 75 J=1,2

```

```

75 ALLINT(I,J)=INTAR(KK,J)
   IF(I.NE.1.AND.ALLINT(I-1,1).EQ.INTAR(KK,1))GO TO 70
   IF(ARCINT(K,2).GE.INTAR(KK,1))GO TO 70
   IF(ICOUN.GT.0)GO TO 70
72 DO 73 L=1,2
73 ALLINT(I,L)=ARCINT(K,L+1)
   IF(K.EQ.K1)ICOUN=1
   K=K+1
   IF(K.GT.K1)K=K1
   KK=KK-1
70 IF(PRINT.EQ.1)WRITE(12,8051)I,(ALLINT(I,J),J=1,2),K,KK
   WRITE(*,8051)I,(ALLINT(I,J),J=1,2),K,KK
8051 FORMAT(T5,'I=',I3,2X,2F12.3,2X,'K=',I3,2X,'KK=',I3)
   IF(PRINT.EQ.1)WRITE(12,8052)
   WRITE(*,8052)
8052 FORMAT(//,T5,'THE APPLICABLE ARRAY ARCINT FOLLOWS:')
   LAL=0
   DO 77 I=1,LL
   R2=SQRT((CX-ALLINT(I,1))**2+(CY-ALLINT(I,2))**2)
   IF(R2.GT.(R+SMLNO))GO TO 77
   LAL=LAL+1
   DO 78 K=1,2
78 ARCINT(LAL,K)=ALLINT(I,K)
   IF(PRINT.EQ.1)WRITE(12,8051)LAL,(ARCINT(LAL,J),J=1,2),I,LAL
   WRITE(*,8051)LAL,(ARCINT(LAL,J),J=1,2),I,LAL
77 CONTINUE
C   THE ARRAY ARCINT HAS BEEN REDEFINED. IT NOW CONTAINS ALL THE
C   APPLICABLE INTERSECTION POINTS IN INCREASING X VALUE.
C   CALCULATE THE WIDTH BETWEEN EACH INTERSECTION POINT.
33 SLICX(1)=S1
   SLIC(1,1,1)=S2
   SLIC(1,1,2)=LINE1
C   DIVIDE WIDTH(L) INTO SLICES OF SIZE NOT GREATER THAN SWIDTH
C   SET SLICE ARRAY SIZE NOSLIC X NO OF LINES & X,Y COORDINATE
C   IF(PRINT.EQ.1)WRITE(7,8057)
   WRITE(*,8057)
8057 FORMAT(//,T5,'FIND SLICE WIDTH AND NO OF SLICES')
98 N=1
   NOSLIC=1
   KM=1
   K=LAL-1
   DO 45 L=1,K
   MM=1
   IF((ARCINT(L+1,1)-ARCINT(L,1)).LE.SWIDTH)GO TO 46
   DO 47 MM=1,100
   AM=MM
   WIDTH=(ARCINT(L+1,1)-ARCINT(L,1))/AM
   IF(WIDTH.LE.SWIDTH)GO TO 49
47 CONTINUE
46 WIDTH=ARCINT(L+1,1)-ARCINT(L,1)
49 NOSLIC=NOSLIC+MM
C   CONTROL FOR DIMENSION OF 75 SPACES IN SLICE ARRAY
   IF(NOSLIC.LT.DIMEN)GO TO 99
   SWIDTH=SWIDTH+0.5
   IF(PRINT.EQ.1)WRITE(12,9999)SWIDTH

```

```

WRITE(*,9999)SWIDTH
9999 FORMAT('O',T5,'***** MAXIMUM SLICE WIDTH HAS BEEN INCREMENTED TO'
1,F5.2,1X,A2)
GO TO 98
99 NSM1=NOSLIC-1
DO 51 I=N,NOLE
IF(LNU(I).EQ.I)GO TO 51
101 DO 52 JJ=KM,NSM1
SLICX(JJ+1)=SLICX(JJ)+WIDTH
SLIC(JJ+1,1,1)=SLIC(JJ,1,1)+WIDTH*SLOPE(I)
52 SLIC(JJ+1,1,2)=I
DIFF=SLICX(JJ+1)-C(I,5)
IF(ABS(DIFF)-0.010)50,50,48
50 N=I+1
48 KM=NOSLIC
GO TO 45
51 CONTINUE
45 CONTINUE
C COMPLETE SLICE ARRAY--CHECK INTERNAL SOIL LINES--SLICE ARRAY SORT
C FOR DECREASING Y-COORDS. *LINE NOT INTERSECTED GIVEN SAME COORDS
NOLP1=NOL+1
NOLEP1=NOLE+1
DO 60 I=1,NOSLIC
N=2
ARCY=CY-SQRT(R**2-(CX-SLICX(I))**2)
DO 59 J=NOLEP1,NOL
IF(LNU(J).EQ.J)GO TO 59
SLIC(I,N,2)=J
SLIC(I,N,1)=C(J,4)+(SLICX(I)-C(J,3))*SLOPE(J)
IF(SLICX(I).LT.(C(J,3)-SMLNO).OR.SLICX(I).GT.(C(J,5)+SMLNO))SLIC(
1I,N,1)=-10.
IF(SLIC(I,N,1).GT.(SLIC(I,1,1)+SMLNO).OR.SLIC(I,N,1).LT.(ARCY-
ASMLNO))SLIC(I,N,1)=-10.0
57 N=N+1
59 CONTINUE
SLIC(I,N,1)=ARCY
60 SLIC(I,N,2)=NOLP1
C SORT SLICE ARRAY IN DECREASING ORDER.
IF(PRINT.NE.1)GO TO 3008
IF(LIST.NE.0)WRITE(12,2116)
3008 IF(LIST.NE.0)WRITE(*,2116)
2116 FORMAT(T6,'SLICE #',2X,'X-COORD',4(2X,'LINE NO',2X,'Y-COORD'))
MCOUN=N
N=MCOUN-1
DO 81 KZ=1,NOSLIC
84 NUM=1
DO 85 KY=1,N
IF(SLIC(KZ,KY,1).LT.-9.50)GO TO 82
IF((SLIC(KZ,KY,1)+SMLNO).GE.SLIC(KZ,KY+1,1))GO TO 85
SAVE=SLIC(KZ,KY,1)
SLIC(KZ,KY,1)=SLIC(KZ,KY+1,1)
SLIC(KZ,KY+1,1)=SAVE
SAVE=SLIC(KZ,KY,2)
SLIC(KZ,KY,2)=SLIC(KZ,KY+1,2)
SLIC(KZ,KY+1,2)=SAVE

```

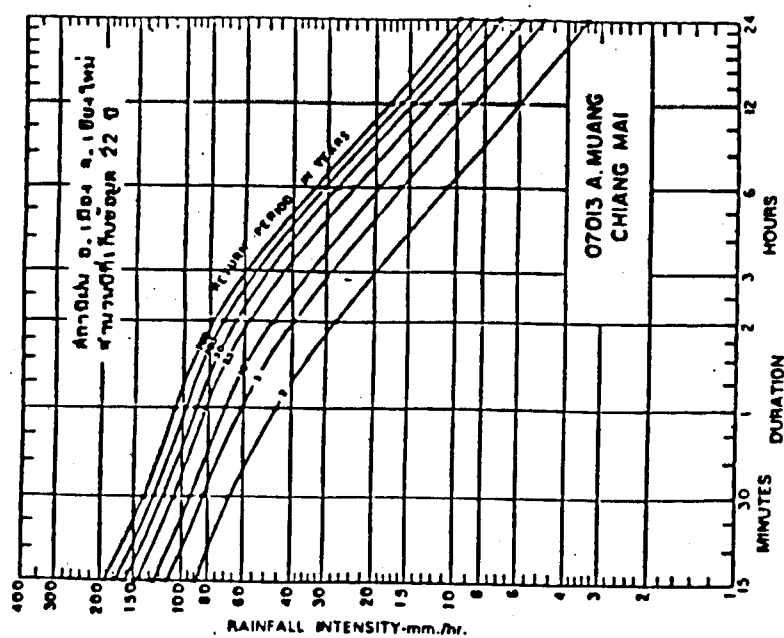
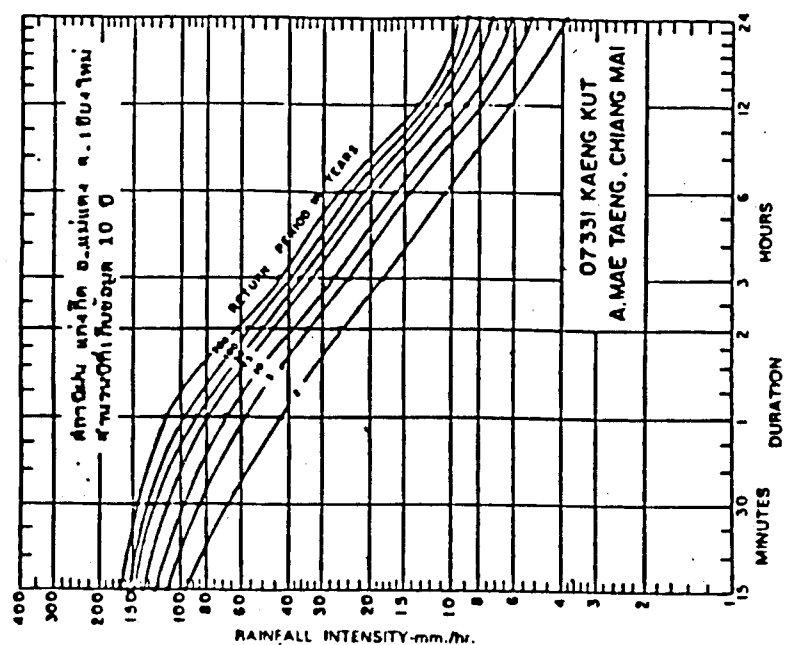
ผลการทดสอบ บ่อดินเยิมที่ใช้ทำตัวเชื่อม

บ่อที่	r_d T/m ³	r_t T/m ³	m	m_{sat}	r_{sat} T/m ³	C_o KN/m ²	C_{sat} KN/m ²	ϕ degree
1	17.46	20.11	0.15	0.20	20.99	97.49	11.77	27.92
2	20.59	22.67	0.10	0.22	25.02	81.39	13.73	29.68

ผลการทดสอบ ฐานรากอ่างเก็บน้ำห้วยแม่ผาแพน

ฐานรากอ่างเก็บน้ำ ห้วยแม่ผาแพน	r_d T/m ³	r_t T/m ³	m	m_{sat}	r_{sat} T/m ³	C_o KN/m ²	C_{sat} KN/m ²	ϕ degree
	16.966	19.171	0.134	0.26	21.377	28.440	13.729	37.596

รูปที่ ค.5 ผลการทดสอบบ่อดินเยิมที่ใช้ทำตัวเชื่อมและฐานราก



รูปที่ ค.6 ความเข้มของฝน-ช่วงเวลา-ความถี่ของฝน จังหวัดเชียงใหม่

ภาคผนวก ง.


```

40 FORMAT(1X,'PLEASE SELECT OPTION AGAIN ??')
   GO TO 1
   ENDIF
   IF(KODE2.EQ.0)GO TO 20
   IF(KODE2.EQ.1)THEN
C
C   COMPUTE DAILY INFLOW HYDROGRAPH BY SCS
C   ADJUST OPTIMUM STORAGE CAPACITY FOR RESERVOIR
C
C   CALL SCS
C   GO TO 1
C   ELSE
C   IF(KODE2.EQ.2)THEN
C
C   COMPUTE & CHECK STABILITY FOR HOMOGENEOUS EARTH DAM
C   MODIFIED & APPLICATION FROM PRINCIPAL 'S A.W. BISHOP
C   BY MR. ARUK SANTAVEESOOK
C
C   CALL STAB
C   GO TO 1
C   ELSE
C   IF(KODE2.EQ.3)THEN
C
C   COMPUTE FLOOD PEAK BY SYNTHESIS UNIT HYDROGRAPH
C   MODIFIED BY SCS (SOIL CONSERVATION SERVICE)
C
C   CALL UNITH
C   GO TO 1
C   ELSE
C   IF(KODE2.EQ.4)THEN
C
C   COMPUTE OUTFLOW HYDROGRAPH FROM RESERVOIR
C   BY FINITE DIFFERENCE
C
C   CALL ROUT
C   GO TO 1
C   ELSE
C   IF(KODE2.EQ.5)THEN
C
C   COMPUTE & DESIGN SPILLWAY
C
C   CALL CHUTE
C   GO TO 1
C   ELSE
C   IF(KODE2.EQ.6)THEN
C
C   COMPUTE FLOW THROUGH EARTH DAM & CHECK UPLIFT PRESSURE
C   BY FINITE DIFFERENCE
C
C   CALL FLOW
C   GO TO 1
C   ENDIF
C   ENDIF
C   ENDIF
C   ENDIF

```

```

        ENDIF
        ENDIF
20 WRITE(*,50)
50 FORMAT(1X,'?? GOOD BYE & SEE YOU AGAIN ?? ')
        STOP
        END
$DEBUG
        SUBROUTINE SCS
C
C      DAILY INFLOW HYDROGRAPH BY SCS (SOIL CONSERVATION SERVICE)
C      MODIFIED BY ARUK SANTAVEESOOK
        DIMENSION OQ(31,12),OU(31,12),QQ(31,12),G(31,12),FLOW(30)
        DIMENSION FF(31,12),FQ(31,12),RAIN(31,12),ET(31,12),MON(12)
        DIMENSION COMAR(10),PERC(10),COEFF(10,12),STOR(30),DELFL(30)
        DIMENSION X11(20),Y11(20),A11(5)
        REAL LL,IA,IM
        COMMON/HYDRO1/OQ,OU,QQ,G,FF,FQ,RAIN,ET
        COMMON/HYDRO2/COMAR,PERC,COEFF
        COMMON/FITT1/STOR,FLOW,DELFL
        COMMON/LST1/X11,Y11,A11
        OPEN(1,FILE='SCS.PRN')
        OPEN(7,FILE='STOR.DAT')
        OPEN(11,FILE='SCS.RES')
        MON(1)=31
        MON(2) = 28
        MON(3)=31
        MON(4)=30
        MON(5)=31
        MON(6)=30
        MON(7)=31
        MON(8)=31
        MON(9)=30
        MON(10)=31
        MON(11)=30
        MON(12)=31
        DO 11 K=1,31
            READ(1,100)RAIN(K,1),RAIN(K,2),RAIN(K,3),RAIN(K,4),RAIN(K,5)
            1,RAIN(K,6)
            1,RAIN(K,7),RAIN(K,8),RAIN(K,9),RAIN(K,10),RAIN(K,11),RAIN(K,12)
            WRITE(*,100)RAIN(K,1),RAIN(K,2),RAIN(K,3),RAIN(K,4),RAIN(K,5)
            1,RAIN(K,6)
            1,RAIN(K,7),RAIN(K,8),RAIN(K,9),RAIN(K,10),RAIN(K,11),RAIN(K,12)
11        CONTINUE
            DO 12 K=1,31
                READ(1,100)ET(K,1),ET(K,2),ET(K,3),ET(K,4),ET(K,5),ET(K,6)
                1,ET(K,7),ET(K,8),ET(K,9),ET(K,10),ET(K,11),ET(K,12)
                WRITE(*,100)ET(K,1),ET(K,2),ET(K,3),ET(K,4),ET(K,5),ET(K,6)
                1,ET(K,7),ET(K,8),ET(K,9),ET(K,10),ET(K,11),ET(K,12)
12        CONTINUE
100       FORMAT(12F6.1)
            READ(1,120) AREA,LL,SV,CPAN,E
            WRITE(*,120)AREA,LL,SV,CPAN,E
120       FORMAT(5F6.2)
C           AREA = WATERSHED AREA.
C           LL =   LONGEST OF STREAM

```

```

C      LC =   THE LENGTH POINT TO THE CENTROID OF AREA
C      SV =   SLOPE OF WATERSHED IN PERCENT
      AMC = RAIN(1,1)+RAIN(2,1)+RAIN(3,1)+
$RAIN(4,1)+RAIN(5,1)
      CA = 25.
      CB = 55.
      CC = 78.
      CD = 72.
      GA = 64.
      GB = 75.
      GC = 83.
      GD = 87.
      EA = 75.
      EB = 84.
      EC = 88.
      ED = 91.
      SA = 98.
      READ(1,1110) WA,WB,WC,WD,AA,AB,AC,AD
      WRITE(*,1110)WA,WB,WC,WD,AA,AB,AC,AD
      READ(1,1110) BA,BB,BC,BD,RA
      WRITE(*,1110)BA,BB,BC,BD,RA
1110  FORMAT(8F6.1)
      READ(1,130) K4
      WRITE(*,130) K4
130   FORMAT(I6)
      DO 140 J1=1,K4
      READ(1,150)COMAR(J1),PERC(J1)
      READ(1,160)COEFF(J1,1),COEFF(J1,2),COEFF(J1,3),COEFF(J1,4),
1COEFF(J1,5),COEFF(J1,6),COEFF(J1,7),COEFF(J1,8),COEFF(J1,9)
2,COEFF(J1,10),COEFF(J1,11),COEFF(J1,12)
      WRITE(*,150)COMAR(J1),PERC(J1)
      WRITE(*,160)COEFF(J1,1),COEFF(J1,2),COEFF(J1,3),COEFF(J1,4),
1COEFF(J1,5),COEFF(J1,6),COEFF(J1,7),COEFF(J1,8),COEFF(J1,9)
2,COEFF(J1,10),COEFF(J1,11),COEFF(J1,12)
150   FORMAT(2F6.2)
160   FORMAT(12F6.2)
140   CONTINUE
      READ(1,13)N1,M1
      WRITE(*,13)N1,M1

C
C      READ STORAGE  =X11(I) *1000 m^3
C      READ ELEVATION =Y11(I)      m
C      N1=NUMBER OF DATA -1
C      M1=DEGREE OF POLYNOMIAL MAX. =5
      NP1=N1+1
      READ(1,14) (X11(I),Y11(I),I=1,NP1)
      WRITE(*,14)(X11(I),Y11(I),I=1,NP1)
13   FORMAT(2I6)
14   FORMAT(2F12.2)
      WX = ((WA*CA) + (WB*CB) + (WC*CC) + (WD*CD))/AREA
      A  = ((AA*GA) + (AB*GB) + (AC*GC) + (AD*GD))/AREA
      B  = ((BA*EA) + (BB*EB) + (BC*EC) + (BD*ED))/AREA
      R  = (RA*SA)/AREA
      CN2 = WX+A+B+R
      IF(AMC.LT.12.) THEN

```

```

CN1 = CN2/(2.281-0.01281*CN2)
CN = CN1
ELSE
IF(AMC.GT.38.) THEN
CN3 = CN2/(0.427+0.00573*CN2)
CN = CN3
ELSE
CN = CN2
ENDIF
ENDIF
CN1 = CN2/(2.281-0.01281*CN2)
P = 0.
S = 25400./CN-254.
SSAT = 25400./CN1-254.
SFC = 0.1*SSAT
HSM = SSAT-S
HG = HSM*2.
HOQ = AMC/3.
IA = 0.2*S
ST = 1000./CN1-10.
TS = (1./27360.)*((3281.*LL)**0.8)*
$((ST+1.)**0.7)/(SV**0.5)
WRITE(11,210)
WRITE(*,210)
210 FORMAT(2X,'DAILY RUNOFF INFLOW HYDROGRAPH',/,2X,'MONTH ',5X,
1'DATE ',10X,'RUNOFF ',8X,'CUMULATIVE ',5X,'CONSUMPTIVE',/,2X
2,',
33',/,)
CMD*10^3 CMD*10^3 CMD*10^
DO 2240 J=1,12
IW = MON(J)
IF (J.EQ.1) THEN
K1=6
ELSE
K1=1
ENDIF
DO 2230 K =K1,IW
P = P + RAIN(K,J)
IF ((P.EQ.0.).OR.(P.LE.IA)) THEN
Q = 0.
F = 0.
ELSE
Q = ((P-IA)**2.)/(P-IA+S)
F = (P-IA)*S/(P-IA+S)
ENDIF
FF(K,J) = F
OQ(K,J) = Q
IF ((J.EQ.1).AND.(K.EQ.6)) THEN
QQ(K,J) = HOQ/6.
GO TO 2222
ENDIF
KK = K-1
JJ = J
IF (KK.EQ.0) THEN
JJ = J-1
KK = MON(JJ)

```

```

SUM2=SUM2+SUM1
230 CONTINUE
HSM = HSM-REC-((ET1+ET2+ET4)/AREA+ET3)
IF(HSM.LT.SFC) THEN
HSM = SFC
ENDIF
G(K,J) = 0.003*HG*(EXP(-(HSM/SSAT)))
HG = HG+REC-G(K,J)
OU(K,J) = QQ(K,J)+G(K,J)
OU(K,J) = OU(K,J)*AREA*E
CUM=CUM+OU(K,J)
IF(J.NE.2)GO TO 41
IF((K.EQ.15).AND.(K.EQ.28)) GO TO 42
41 IF((K.NE.15).AND.(K.NE.30))GO TO 47
42 L1=L1+1
FLOW(L1)=CUM
47 WRITE (11,200) J,K,OU(K,J),CUM,SUM2
WRITE(*,200)J,K,OU(K,J),CUM,SUM2
200 FORMAT (2X,I2,9X,I2,8X,F12.2,5X,F12.2,5X,F12.2)
2230 CONTINUE
2240 CONTINUE
C
C FIND OPTIMUM STORAGE CAPACITY
C
DO 43 NN=1,L1
DO 44 MM=1,NN
DELFL(MM)=FLOW(NN)*MM/NN-FLOW(MM)
WRITE(*,46)DELFL(MM)
46 FORMAT(1X,' STORAGE m^3 =',F12.3)
44 CONTINUE
XMAX=DELFL(1)
DO 45 I=2,MM
IF(XMAX-DELFL(I)) 430,45,45
430 XMAX=DELFL(I)
45 CONTINUE
STOR(NN)=XMAX
43 CONTINUE
XMAX=STOR(1)
DO 51 I=2,NN
IF(XMAX-STOR(I)) 301,51,51
301 XMAX=STOR(I)
51 CONTINUE
CALL LST(X11,Y11,N1,M1,A11)
ELV=A11(1)+A11(2)*XMAX+A11(3)*(XMAX**2)
WRITE(11,102)XMAX,ELV
WRITE(*,102)XMAX,ELV
102 FORMAT(1X,'OPTIMUM STORAGE =',F12.3,'*1000 m^3',5X,'ELEVATION =',
1F12.3,' m ')
WRITE(7,103)NP1
103 FORMAT(I12)
WRITE(7,104)(X11(I),Y11(I),I=1,NP1)
104 FORMAT(2F12.2)
CLOSE(1)
CLOSE(7)
CLOSE(11)

```

```

ENDIF
FQ(K,J) = OQ(K,J)-OQ(KK,JJ)
IF (FQ(K,J).LT.0.) THEN
FQ(K,J) = 0.
ENDIF
HOQ = HOQ+FQ(K,J)
IM = (FQ(K,J)+FQ(KK,JJ))/2.
QQ(K,J) = QQ(KK,JJ)+((IM-QQ(KK,JJ))/(TS+0.5))
F = FF(K,J)-FF(KK,JJ)
IF (F.LT.0.) THEN
F = 0.
ENDIF
IF (QQ(K,J).LT.0.0) THEN
QQ(K,J) = 0.
ENDIF
IF(QQ(K,J).GT.HOQ) THEN
QQ(K,J) = HOQ
ENDIF
2222 HOQ = HOQ-QQ(K,J)
HSM = HSM+F
XX = HSM/SSAT
IF(HSM.GT.SFC) THEN
REC = XX*(HSM-SFC)*0.106
ELSE
REC = 0.
ENDIF

C
C COMPUTE COMSUMPTIVE USED AND IRRIGATION REQUIREMENT
C EFFT=effective rainfall mm/day
C PERC(J1)=percolation for command area(j1) mm/day
C RAIN(K,J)=precipitation mm/day
C 0.65=conveyance efficiency
C
EVAP=CPAN*ET(K,J)
W1=WA+WB+WC+WD
A2=AA+AB+AC+AD
B2=BA+BB+BC+BD
ET1=W1*0.75*EVAP
ET2=A2*0.89*EVAP
ET3=B2*(XX*EVAP)/AREA
ET4=RA*ET(K,J)
DO 230 J1=1,K4
IF(COEFF(J1,J).EQ.0) THEN
EFFT=0.
ELSE
EFFT=(1-0.006*RAIN(K,J))*RAIN(K,J)
IF(EFFT.LT.0)THEN
EFFT=0.
ENDIF
ENDIF
SUM1=COMAR(J1)*(COEFF(J1,J)*ET(K,J)-EFFT+PERC(J1))
IF(SUM1.LT.0)THEN
SUM1=0
ENDIF
SUM1=SUM1/0.65

```

```

      RETURN
      END
$DEBUG
      SUBROUTINE STAB
C
C      SLOPE STABILITY PROBLEM BY SIMPLIFIED BISHOP METHOD
C      READ TOP INTERSECTION OF VERTICAL LINE FIRST THEN OTHER END
C      READ LINE INTERSECTION NUMBERS WITH INCREASING X-COORD.
C      NOL = NO OF SOIL LINES--NUMBER EXTERNAL LINES FIRST FROM LT TO RT
C      NLIT = TOTAL NUMBER OF LINE INTERSECTIONS.
C      NOS = NUMBER OF SOILS INCLUDED IN SECTION--USE ANY ORDER
C      NOLE = NUMBER OF EXTERNAL SOIL LINES.
C      DIMEN = CONTROL FOR NUMBER OF SLICES ALLOWED AS 75, 80, 90, ETC
C      C(I,J) = LINE COORD MATRIX--I = LINE NO., J=2 IS NO OF INTERSECT
C              ON LINE, J=3-6 ARE X1,Y1,X2,Y2
C      READ LINE END COORDS X1,Y1,X2,Y2 LEFT TO RIGHT
C      ITX,ITY = # OF CIRCLE CENTER POINTS IN X,Y DIRECTIONS.
C      LIST IF GREATER THAN 0 WILL LIST ADDITIONAL DATA
C      SWIDTH = INIT. MAX SLICE WIDTH--AUTO CHANGED IF EXCEEDS "DIMEN"
C      SCALE = PLOT SCALE 1" = 10' USE 10.; 1" = 20' USE 20., ETC
C      IF PCODE = 0 THE PLOT ROUTINE IS SKIPPED
C      NP IS A PLOT CONTROL COUNTER; IF NP = 3 EVERY THIRD ONE IS PLOTTED
C      IF A LINE TERMINATES AT A SOIL JOINT, IT IS INCL BUT BOTH INTERSEC
C      ARE THOSE OF THE JOINT
C      SAT = SWITHC FOR SATURATED SOIL IF SAT >0 SOIL SATURATED
C      UT1 = FT OR M FOR SCALE; UT2 = IN OR CM FOR SCALE; READ COORDS A
C      UNIT WEIGHT IN US OR METRIC UNITS--TITLE TO GIVE UNITS
C
      DIMENSION C(15,6),NOLIT(15,10),SLOPE(15),INTAR(15,2),EFFWT(100),
      2NSLIN(10),G(10),PHI(10),COHES(10),SLICX(76),SOIL(8,9,4),SAT(10),
      3ARCINT(20,3),LNU(15),CO(100),ALPHA(100),TITLE(20),SLIC(76,10,2),
      4P(100),B(100),AREA(100),WEIGH(100),ALLINT(30,3),FA(100),FO1(100)
      COMMON/HYDRO3/C,NOLIT,INTAR,ALLINT,SLIC,ARCINT
      COMMON/HYDRO4/EFFWT,CO,ALPHA,P,B,AREA,WEIGH,FA,FO1
      COMMON/HYDRO5/NSLIN,G,PHI,COHES,SAT,SLICX
      INTEGER PCODE,DIMEN,PRINT
      REAL INTAR,METER
      CHARACTER*80 HEAD
      OPEN(2,FILE='STAB.PRN')
      OPEN(12,FILE='STAB.RES')
      BIGNO = 9999999.
      SMLNO = 0.01
      READ(2,7000)HEAD
7000 FORMAT(//////////////////////////////////////
1////////A80)
4000 READ(2,1000,END=150)TITLE,UT1,UT2,FU4
C
C $$$ DATA INPUT $$$
C      TITLE = EXPLANATION FOR PROJECT
C      UT1 = SCALE IF SI UNIT USE M BUT IF ELS UNIT USE FT
C      UT2 = SCALE IF SI UNIT USE CM BUT IF ELS UNIT USE IN
C      FU4 = GRAVITY (g) IF SI UNIT USE 9.807 BUT IF ELS UNIT USE 32.2
C
      WRITE(*,500)
      WRITE(12,500)

```

```

500 FORMAT(10X,'*****')
    WRITE(*,501)
    WRITE(12,504)
    WRITE(12,501)
    WRITE(12,502)
    WRITE(12,503)
501 FORMAT(10X,'*   CALCULATED BY   MR. ARUK   SANTAVEESOOK   *')
502 FORMAT(10X,'*                               WATER RESOURCE ENGINEERING   *')
503 FORMAT(10X,'*                               KHON KAEN UNIVERSITY   *')
504 FORMAT(10X,'*                               *')
    WRITE(12,504)
    WRITE(*,500)
    WRITE(12,500)
    WRITE(*,1000)TITLE,UT1,UT2,FU4
1000 FORMAT(20A4/A4,6X,A4,6X,F8.4)
    WRITE(12,7001)HEAD
7001 FORMAT(A80)
    WRITE(12,2000)TITLE
    WRITE(*,2000)TITLE
2000 FORMAT('1',T5,20A4,/)
    READ(2,123)NOL,NLIT,NOS,NOLE,ITX,ITY,PCODE,NP,DIMEN,LIST
C
C $$ DATA INPUT $$
C   NOL =TOTAL NUMBER OF SOIL LINE
C   NLIT=TOTAL NUMBER OF LINE INTERSECTIONS (the end of any line whether
C         or not intersected by another line is a line intersection)
C   NOS =NUMBER OF SOIL IN MASS (same soil submerged is counted twice)
C   NOLE=NUMBER OF TOP EXTERNAL SOIL LINES
C   ITX,ITY = NUMBER OF CIRCLE CENTER POINTS IN X,Y DIRECTIONS TO BE
C             ANALYZED FOR A SINGLE ENTRANCE POINT
C   DIMEN = CONTROL NUMBER OF SLICE AS 75,80,90 etc
C
    WRITE(*,123)NOL,NLIT,NOS,NOLE,ITX,ITY,PCODE,NP,DIMEN,LIST
123 FORMAT(I7,9I8)
    READ(2,1010)CX,CY,ENTX,ENTY,DELX,DELY,SWIDTH,SCALE
C
C $$ DATA INPUT $$
C   CX & CY   = INITIAL TRIAL CENTER COORDINATES
C   ENTX & ENTY = TRIAL CIRCLE ENTRANCE COORDINATES
C   DELX & DELY = CENTER X & Y COORDINATE INCREMENTS FOR EACH TRIAL
C   SWIDTH     = INITIAL SLICE WIDTH
C
    WRITE(*,1010)CX,CY,ENTX,ENTY,DELX,DELY,SWIDTH,SCALE
1010 FORMAT(F7.3,7F8.3)
    WHOLD = SWIDTH
    NOSPI=NOS+1
    WRITE(12,2001)NOL,NLIT,NOS,NOLE,ITX,ITY,SWIDTH,UT1
2001 FORMAT(T5,'NO OF LINES =',I3,5X,'NO OF LINE INTERSECT =',I3,/,T5,
A'NO OF SOILS =',I3,5X,'NO OF EXTERNAL SOIL LINES =',I3,/,T5,
A'NO OF X-INCREMENTS =',I3,5X,'NO OF Y-INCREMENTS =',I3,/,T10,'IN
2ITIAL SLICE WIDTH =',F5.1,1X,A2/)
    WRITE(12,2003)
2003 FORMAT(T5,'THE LINE END COORD MATRIX',/,T4,'LINE NO',T12,'NO INT',
1T21,'X1',8X,'Y1',8X,'X2',8X,'Y2',6X,'SLOPE',5X,'LINE INT NO')
    DO 333 I=1,NOL

```

```

C
C      NLI = NO OF INTERSECT ON ANY LINE --MUST HAVE AT LEAST 2-VALUES
      READ(2,123)NLI
      READ(2,1012)(C(I,J),J=1,6),(NOLIT(I,N),N=1,NLI)
C
C  $$ DATA INPUT $$
C      C(I,J) = LINE DATA INCLUDING LINE NUMBER
C              NUMBER OF LINE INTERSECTIONS FOR THE LINES, THE X,Y CODINATE
C              OF THE END POINTS LEFT TO RIGHT(SIX ENTRIES )
C      NOLIT(I,N) = ALL THE LINE INTERSECTION NUMBER ON THE i th LINE
C                  INCLUDING THE END VALUES (NLI entries)
C
      WRITE(*,123)NLI
      WRITE(*,1012)(C(I,J),J=1,6),(NOLIT(I,N),N=1,NLI)
1012  FORMAT(F7.1,F8.1,4F8.3,3I8)
      SLOPE(I) = BIGNO
      IF(ABS(C(I,5)-C(I,3)).LE.0.001)GO TO 333
      SLOPE(I)=(C(I,6)-C(I,4))/(C(I,5)-C(I,3))
333  WRITE(12,2004)(C(I,J),J=1,6),SLOPE(I),(NOLIT(I,KK),KK=1,NLI)
      WRITE(*,2004)(C(I,J),J=1,6),SLOPE(I),(NOLIT(I,KK),KK=1,NLI)
2004  FORMAT(T6,F3.0,3X,F3.0,T15,4(2X,F8.2),G13.6,1X,6I5)
C
C      READ X AND Y-COORDINATES TO BUILD LINE INTERSECT ARRAY (INTAR)
C
      WRITE(12,2005)
2005  FORMAT(/,T5,'LINE INTERSECT ARRAY',/,T4,'INT NO',T16,'X',T28,'Y')
      DO 2 J=1,NLIT
      READ(2,1010)(INTAR(J,K),K=1,2)
      WRITE(*,1010)(INTAR(J,K),K=1,2)
2    WRITE(12,2006)J,(INTAR(J,K),K=1,2)
2006  FORMAT(T6,I3,T12,F10.2,2X,F10.2)
C
C      BUILD SOIL INTERSECTION ARRAY USING LINE INTERSECT NO. (INTNO)
C
      WRITE(12,2008)
2008  FORMAT(/,T5,'SOIL DATA ARRAY',/,T4,'SOIL NO',T13,'LINE #',T21,
A'LEFT INT',3X,'RT. INT',3X,'SAT',3X,'UNIT WT',3X,'PHI',3X,'COHESI
2ON')
C
C  $$ DATA INPUT $$
C      NUMBER SOILS IN ANY ORDER =====> REFERABLY FROM LEFT TO RIGHT
C      NSLIN(I) = NUMBER OF SOIL LINES DEFINING THE BOUNDARY OF THE SOIL.
C                INCLUDE LINES TERMINATING AT A JOINT. IF A LINES
C                INTERSECTS A SOIL-LINE BOUNDARY BETWEEN THE ENDS,COUNT
C                THE SOIL BOUNDARY LINE TWICE
C      G(I) = UNIT WEIGHT EITHER SATURATED OR WET
C      PH(I) = 0 ANGLE
C      COHES(I) = COHESION
C      SAT = 1. IF SATURATED
C           = 0. IF WET
C
      DO 5 I=1,NOS
      READ(2,1016)NSLIN(I),G(I),PHI(I),COHES(I),SAT(I)
      WRITE(*,1016)NSLIN(I),G(I),PHI(I),COHES(I),SAT(I)
1016  FORMAT(I7,4F8.3)

```

```

      NS=NSLIN(I)
      DO 5 K = 1,NS
      READ(2,123)LINSOL,INTL,INTR
C
C   DATA INPUT
C       LINSOL = SOIL LINE NUMBER
C       INTL & INTR = INTERSECTION NUMBER ON LEFT & RIGHT END OF LINE
C                   (IF A SOIL LINE TERMINATES AT A JOINT ON A SOIL BOUNDARY,
C                   THAT LINE IS INCLUDED ; THE JOINT NUMBER IS USED FOR BOTH
C                   INTL & INTR)
C
      WRITE(*,123)LINSOL,INTL,INTR
      SOIL(I,K,1)=LINSOL
      SOIL(I,K,2)=INTL
      SOIL(I,K,3)=INTR
      SOIL(I,K,4) = SAT(I)
5  WRITE(12,2009)I,(SOIL(I,K,MM),MM=1,4),G(I),PHI(I),COHES(I)
2009 FORMAT(T5,I3,6X,F3.0,7X,F3.0,6X,F3.0,6X,F2.0,5X,F6.1,3X,F4.1,3X
      A,F7.1)
C
C   DATA INPUT FOR SURCHARGE & EXTERNAL LOAD
C       BEAR=LOAD PER UNIT WIDTH          kN/m
C       CL=LENGTH OF SURCHARGE & EXTERNAL LOAD  m
C   CONDIAION ??
C       KSUR = 0 NO SURCHARGE & EXTERNAL LOAD
C       = 1 CASE BEGINNING OF DREDGING
C
      READ(2,15)KSUR
      WRITE(*,15)KSUR
15  FORMAT(I7)
      IF(KSUR.NE.0)THEN
      READ(2,17)BEAR,CL,EDGX,EDGY
      WRITE(*,17)BEAR,CL,EDGX,EDGY
17  FORMAT(F15.3,3F8.2)
      ENDIF
      KFO = 0
3004 PCOUN=0.
      IF(PRINT.EQ.1)GO TO 3020
      CY1=CY
      CX1=CX
3020 CY=CY1
      CX=CX1
C   BEGIN LOOP TO TEST TRIAL CIRCLES
100 DO 360 IY=1,ITY
      IF(IY.GT.1)CY=CY+DELY
      DO 360 IX=1,ITX
      PCOUN = PCOUN + 1.
      NCOUN = PCOUN
      SWIDTH = WHOLD
      IF(IX.GT.1)CX=CX+DELX
      IF(KFO.EQ.NCOUN)GO TO 3005
      IF(PRINT.EQ.1)GO TO 360
C   COMPUTE RADIUS OF TRIAL CIRCLE TEST EACH LINE FOR INTERSECT WITH
C   TRIAL RADIUS---CX,CY ARE INITIAL TRIAL CENTER OF ARC COORDINATE
3005 R=SQRT((ENTX-CX)**2+(CY-ENTY)**2)

```

```

IF(PRINT.EQ.1)WRITE(12,2121)NCOUN,CX,CY,ENTX,ENTY,R
WRITE(*,2121)NCOUN,CX,CY,ENTX,ENTY,R
2121 FORMAT(/,T5,'TRIAL CIRCLE NO =',I3,/,T5,'CIRCLE CTR COORDS:',2X,
1'X=',F10.2,2X,'Y=',F10.2,/,T5,'ENTRANCE PT. COORDS:',2X,'X=',F10
2.2,2X,'Y=',F10.2,/,T10,'TRIAL ARC RADIUS=',F10.3,/)
K1=0
DO 8 I=1,NOL
LNU(I)=0
IF(ABS(SLOPE(I)).LE.0.0001)GO TO 9
CON=C(I,3)-C(I,4)/SLOPE(I)
AA=1.0/SLOPE(I)**2+1.0
BB=2.0*CON/SLOPE(I)-2.0*CX/SLOPE(I)-2.0*CY
CC=CON**2-2.*CX*CON+CX**2+CY**2-R**2
DIFF=BB**2-4.0*AA*CC
IF(DIFF.LT.0.0)GO TO 20
YPR=(-BB+SQRT(DIFF))/(2.0*AA)
YNR=(-BB-SQRT(DIFF))/(2.0*AA)
C THIS PART COMPUTES X-COORDS IF CIRCLE INTERSECTS ANY BUT
C HORIZONTAL LINES (SLOPE(I)>0.0001).
XPR=YPR/SLOPE(I)+CON
XNR=YNR/SLOPE(I)+CON
GO TO 10
C FOLLOWING STEPS USED FOR HORIZONTAL LINES
9 DIFF=R**2 - (CY-C(I,4))**2
IF(DIFF.LT.0.)GO TO 20
XPR=CX+SQRT(DIFF)
XNR=CX-SQRT(DIFF)
YPR=C(I,4)
YNR=C(I,4)
10 J1=0
J2=0
IF(ABS(SLOPE(I)).GE.BIGNO)GO TO 11
IF(XPR.GE.C(I,3).AND.XPR.LE.C(I,5))J1=1
IF(XNR.GE.C(I,3).AND.XNR.LE.C(I,5))J2=1
GO TO 12
11 IF(SLOPE(I))66,66,666
66 IF(YPR.GE.C(I,6).AND.YPR.LE.C(I,4))J1=1
IF(YNR.GE.C(I,6).AND.YNR.LE.C(I,4))J2=1
GO TO 12
666 IF(YPR.GE.C(I,4).AND.YPR.LE.C(I,6))J1=1
IF(YNR.GE.C(I,4).AND.YNR.LE.C(I,6))J2=1
12 IF(J2.EQ.0)GO TO 13
K1=K1+1
ARCINT(K1,1)=I
ARCINT(K1,2)=XNR
ARCINT(K1,3)=YNR
13 IF(J1.EQ.0)GO TO 7
K1=K1+1
ARCINT(K1,1)=I
ARCINT(K1,2)=XPR
ARCINT(K1,3)=YPR
GO TO 8
7 IF(J1.NE.0.OR.J2.NE.0)GO TO 8
C END OF LOOP FOR ARC & LINE INTERSECTIONS WITH ARC INT.ARRAY BUILT
20 LNU(I)=I

```

```

      GO TO 85
82  SLIC(KZ,KY,1)=SLIC(KZ,KY-1,1)
      SLIC(KZ,KY,2)=SLIC(KZ,KY-1,2)
85  IF(KY.NE.1.AND.SLIC(KZ,KY,1)-SMLNO.LE.SLIC(KZ,KY-1,1))NUM=NUM+1
      IF(NUM.NE.N)GO TO 84
      IF(PRINT.NE.1)GO TO 3009
      IF(LIST.NE.0) WRITE(12,3) KZ,SLICX(KZ),(SLIC(KZ,KY,2),
1SLIC(KZ,KY,1),KY=1,MCOUN)
3009 IF(LIST.NE.0) WRITE(*,3) KZ,SLICX(KZ),(SLIC(KZ,KY,2),
1SLIC(KZ,KY,1),KY=1,MCOUN)
81  CONTINUE
      3  FORMAT(T6,I5,9F9.2,/,T20,8F9.2,/,T20,8F9.2,/,T20,8F9.2)
      DO 306 I=1,NSM1
      SAREA=0.0
      WEIGHT=0.0
      EFWT=0.
      ISOIL=0
      NN=MCOUN-1
      IF(PRINT.EQ.1)GO TO 3010
      IF(LIST.NE.0)WRITE(12,350)I
3010 IF(LIST.NE.0)WRITE(*,350)I
      350 FORMAT(/,T5,'SLICE LINE NUMBER',I4)
      DO 303 J=1,NN
      DA=(SLIC(I,J,1)+SLIC(I+1,J,1)-SLIC(I,J+1,1)-SLIC(I+1,J+1,1))*
A(SLICX(I+1)-SLICX(I))/2.0
      IF(DA.LE.SMLNO)GO TO 303
      DO 305 II=1,NOSP1
      IF(II.EQ.NOSP1)GO TO 600
      IF(ISOIL.EQ.II)GO TO 305
      N=NSLIN(II)
      ICOUNT=0
      JCCOUNT=0
311  DO 304 JJ=1,N
      IF(JCOUNT.EQ.2)GO TO 305
      INTL=SOIL(II,JJ,2)
      INTR=SOIL(II,JJ,3)
      IF(ICOUNT.EQ.1)GO TO 310
      IF(SLIC(I,J,2).NE.SOIL(II,JJ,1))GO TO 304
      ICOUNT=1
      JSOIL=II
      IF((SLICX(I)+SMLNO).GE.INTAR(INTL,1).AND.(SLICX(I)-SMLNO).LE.
AINTAR(INTR,1))GO TO 310
      ICOUNT=0
      GO TO 304
310  IF(SLIC(I+1,J,2).NE.SOIL(II,JJ,1))GO TO 304
      IF((SLICX(I+1)+SMLNO).LT.INTAR(INTL,1).OR.(SLICX(I+1)-SMLNO).GT.
AINTAR(INTR,1))GO TO 304
      ICOUNT=2
      IF(JSOIL.NE.II)GO TO 305
302  ISOIL=II
308  SAREA=SAREA+DA
      GSUB=G(ISOIL)
      IF(SAT(ISOIL).GT.0.1)GSUB=G(ISOIL)-FU4
      EFWT=EFWT+DA*GSUB
      WEIGHT=WEIGHT+DA*G(ISOIL)

```

```

      IF(LIST.EQ.0)GO TO 9092
      IF(PRINT.EQ.1)WRITE(12,351)J,I,DA
      WRITE(*,351)J,I,DA
351  FORMAT(T10,'DSLICE NO',I2,'OF SLICE -',I3,'WITH DA OF',F10.3)
      IF(PRINT.EQ.1)WRITE(7,352)ISOIL,J,SAREA,WEIGHT,EFWT
      WRITE(*,352)ISOIL,J,SAREA,WEIGHT,EFWT
352  FORMAT(T15,'SOIL -',I3,'LIES IN DSLICE -',I3,'/',T15,'TOTAL AREA='
      A,F10.3,3X,'TOTAL WEIGHT =',G10.3,5X,'EFFECT. WT =',G10.3)
9092 GO TO 303
304  CONTINUE
      IF(ICOUNT.EQ.1)JCOUNT=JCOUNT+1
      IF(ICOUNT.EQ.1)GO TO 311
      GO TO 305
600  SAREA=SAREA+DA
      GSUB=G(ISOIL)
      IF(SAT(ISOIL).GT.0.1)GSUB=G(ISOIL)-FU4
      EFWT=EFWT+DA*GSUB
      WEIGHT=WEIGHT+DA*G(ISOIL)
      IF(LIST.EQ.0)GO TO 303
      IF(PRINT.EQ.1)WRITE(12,651)J,I,DA
      WRITE(*,651)J,I,DA
651  FORMAT(T10,'DSLICE NO',I2,'OF SLICE-',I3,'WIDTH DA OF',F10.3)
      IF(PRINT.EQ.1)WRITE(12,652)ISOIL,J,SAREA,WEIGHT,EFWT
      WRITE(*,652)ISOIL,J,SAREA,WEIGHT,EFWT
652  FORMAT(T15,'SOIL-',I3,'LIES IN DSLICE-',I3,'/',T15,'TOTAL AREA='
      A,F10.3,3X,'TOTAL WEIGHT=',G10.3,5X,'EFFECT. WT=',G10.3)
      GO TO 303
305  CONTINUE
303  CONTINUE
      ALPHA(I)=ASIN(ABS(CX-((SLICX(I+1)-SLICX(I))/2.0+SLICX(I))))/R)
      AREA(I)=SAREA
      EFFWT(I)=EFWT
      WEIGH(I)=WEIGHT
      CO(I)=COHES(ISOIL)
306  P(I)=PHI(ISOIL)
      IF(LIST.EQ.0)GO TO 365
      IF(PRINT.EQ.1)WRITE(12,354)
      WRITE(*,354)
354  FORMAT(T5,'SLICE #',3X,'AREA',3X,'WEIGHT',4X,'COHESION',3X,'PHI',
      A3X,'ALPHA')
      DO 307 I=1,NSM1
      IF(AREA(I).LE.SMLNO)GO TO 362
      IF(PRINT.EQ.1)WRITE(12,353)I,AREA(I),WEIGH(I),CO(I),P(I),ALPHA(I)
      WRITE(*,353)I,AREA(I),WEIGH(I),CO(I),P(I),ALPHA(I)
      GO TO 307
362  IF(PRINT.EQ.1)WRITE(7,353)I,AREA(I),WEIGH(I)
      WRITE(*,353)I,AREA(I),WEIGH(I)
353  FORMAT(T7,I2,2X,3F10.3,F6.2,F9.4)
307  CONTINUE
C    SUM OVERTURN. MOMENT AND COMPUTE PART OF RESIST. TERM
365  DO 367 I=1,NSM1
      IF(AREA(I).LE.SMLNO)P(I)=0.00
      IF(AREA(I).LE.SMLNO)CO(I)=0.00
367  P(I)=P(I)/57.2958
      FI=1.0

```

```

383 ZUM=0.0
    TF=0.0
    DO 382 K=1,NSM1
        CENTR=(SLICX(K+1)-SLICX(K))/2.0+SLICX(K)
        IF((CENTR-CX).LT.0.0)T=-WEIGH(K)*SIN(ALPHA(K))
        IF((CENTR-CX).EQ.0.0)T=0.0
        IF((CENTR-CX).GT.0.0)T=WEIGH(K)*SIN(ALPHA(K))
394 TF=TF+T
        B(K)=(SLICX(K+1)-SLICX(K))/COS(ALPHA(K))
        Z=(CO(K)*B(K)+EFFWT(K)*TAN(P(K)))/(COS(ALPHA(K))+TAN(P(K))*SIN(AL
1PHA(K))/FI)
382 ZUM=ZUM+Z
C   COMPUTE WITH SURCHARGE AND EXTERNAL LOAD
    IF(KSUR.EQ.0)GOTO 21
    IF((ENTX-EDGX).LT.CL)THEN
        EXTLOAD=BEAR*(ENTX-EDGX)*((EDGX-CX)+(ENTX-EDGX)/2)
    ELSE
        EXTLOAD=BEAR*CL*(EDGX-CX+CL/2)
    ENDIF
21  TF=TF+EXTLOAD/R
    FO=ZUM/TF
    WRITE(*,19)EXTLOAD
    WRITE(12,19)EXTLOAD
19  FORMAT(1X,'    EXTERNAL LOAD =',F15.2,'    kN-m')
    IF(PRINT.EQ.1)GO TO 3018
    WRITE(*,116)FI,FO
116 FORMAT(20X,'FI= ',F10.5,3X,'FO= ',F10.5)
    DEN=0.0
    ZUM=0.0
    DO 384 K=1,NSM1
        F=FO
        Z=(CO(K)*B(K)+EFFWT(K)*TAN(P(K)))/(F*COS(ALPHA(K))+TAN(P(K))*SIN
1(ALPHA(K)))
        ZUM=ZUM+Z
        D=(CO(K)*B(K)+EFFWT(K)*TAN(P(K)))*TAN(P(K))*SIN(ALPHA(K))/(F*COS
1(ALPHA(K))+TAN(P(K))*SIN(ALPHA(K)))*2
384 DEN=DEN+D
        FI=FO*(1.0-(TF-ZUM)/(TF-DEN))
386 IF(ABS(FI-FO)-.001)385,385,383
385 FO1(NCOUN)=FO
        IF(CX.GT.ENTX)FO1(NCOUN)=-FO1(NCOUN)
        KFO=NCOUN
        WRITE(*,108)NCOUN,FO
108 FORMAT('0',15X,'THE SAFETY FACTOR FOR POINT ',I3,' IS ',F10.5)
        IF(PCODE.EQ.0.OR.R.GE.CY)GO TO 360
C   PLOT PLOT PLOT PLOT PLOT PLOT PLOT PLOT PLOT PLOT
360 IF(IX.EQ.ITX)CX=CX-(IX-1)*DELX
        DO 3013 I=1,NCOUN
            FA(I)=FO1(I)
3013 CONTINUE
C
C   CHECK ??
C       TO select the minimum factor safety slope stability
C       for one entrance point
C

```

```

DO 800 I=1,NCOUN
LFA=NCOUN-I
DO 800 J=1,LFA
IF(FA(J)-FA(J+1))801,800,800
801 SAFA=FA(J)
FA(J)=FA(J+1)
FA(J+1)=SAFA
800 CONTINUE
DO 3014 I=1,NCOUN
IF(FA(NCOUN).EQ.FO1(1))GO TO 3015
3014 CONTINUE
3015 KFO=1
PRINT=1
GO TO 3004
3018 WRITE(*,3016)KFO,FO1(KFO)
WRITE(12,3017)
WRITE(12,3016)KFO,FO1(KFO)
3016 FORMAT(/,10X,'*>>THE CRITICAL POINT ',I3,' MINIMUM FACTOR SAFETY I
1S',F8.4,'<<*')
3017 FORMAT(/,10X,'*****')
WRITE(12,3017)
GO TO 4000
CLOSE(2)
CLOSE(12)
150 RETURN
END
$DEBUG
SUBROUTINE CHUTE
C
C DESIGN CHUTE SPILLWAY
C INLET SECTION USE SCS (STAIGTH INLET)
C CHUTE SECTION
C STEEP SECTION
C STILLING BASIN USE USBR
C
C
C DIMENSION DL3(12),DL4(4),FR3(12),FR4(4),DEPTH(20),STA(20),ELV(20)
1,HV(20),SLOPE(20),VEL(20)
C DIMENSION X(100),X1(100),X2(100),X3(100),X4(100),X5(100),ELVB(20)
C DIMENSION Y(100),Y1(100),Y2(100),Y3(100),Y4(100),Y5(100)
C REAL MANCH,MANER
C DATA DL3/2.23,2.45,2.55,2.63,2.7,2.72,2.74,2.75,2.75,2.75,2.74,
12.72/
C DATA DL4/4.3,5.25,5.79,6./
C DATA FR3/4.5,6.,7.,8.,9.,10.,11.,12.,13.,14.,15.,16./
C DATA FR4/2.,3.,4.,5./
C READ DATA INPUT
C QPEAK = MAX. FLOOD PEAK FOR RETURN PERIOD
C WL = OPENNING SPILLWAY
C MANCH = MANNING OF CHUTE
C STA(I) = STATION TO ANY POINT
C ELV(I) = ELEVATION TO ANY POINT
C SLOPE(I)= SLOPE OF CHUTE STA. TO STA.
C VEL(I) = VELOCITY TO ANY STA.

```

```

C      HV(1)   = VELOCITY HEAD TO ANY STA.
C      MANER   = MANNING AT DOWNSTREAM
C      SS      = SIZE SLOPE AT DOWNSTREAM
C      S       = SLOPE AT DOWNSTREAM
C      VELD    = VELOCITY ALLOWED AT DOWNSTREAM
      OPEN(5,FILE='CHUTE.PRN')
      OPEN(9,FILE='CHUTE.DAT')
      OPEN(15,FILE='CHUTE.RES')
      OPEN(19,FILE='DAM.DAT')
      OPEN(21,FILE='AA.SCR')
      OPEN(22,FILE='PLAN.SCR')
      OPEN(23,FILE='DD.SCR')
      OPEN(24,FILE='BB.SCR')
      OPEN(25,FILE='CC.SCR')
      OPEN(26,FILE='DAM.SCR')
      READ(9,37)QPEAK,WL
37  FORMAT(2F12.2)
      READ(5,1)MANCH
      WRITE(*,1)MANCH
1  FORMAT(////////////////////////////////////////F10.4)
      READ(5,4)MANER,S,SS,VELC,FT,ERR
      WRITE(*,4)MANER,S,SS,VELC,FT,ERR
4  FORMAT(2F10.4,2F10.2,2F10.6)
      READ(5,2)NOSTA
      WRITE(*,2)NOSTA
2  FORMAT(110)
      READ(5,3)(STA(I),ELV(I),SLOPE(I),I=1,NOSTA)
      WRITE(*,3)(STA(I),ELV(I),SLOPE(I),I=1,NOSTA)
3  FORMAT(2F10.2,F10.6)
      BCH=WL
C      FIND TAIL WATER DEPTH
11  TR=(WL**2+4*SS*QPEAK)**0.5
      TAIL1=(-WL+TR)/2/SS
      TAIL2=(-WL-TR)/2/SS
      IF(TAIL1.LT.TAIL2)GO TO 5
      TAIL=TAIL1
      GO TO 6
5  TAIL=TAIL2
      WRITE(*,52)TAIL
52  FORMAT(10X,'TAIL =',F6.2)
C      TRIAL DEPTH AT DOWNSTREAM BY BISECTION METHOD
6  G=TAIL
      XT=2
7  RT=(WL+SS*TAIL)*TAIL/((WL+2*TAIL*((1+SS**2)**0.5)))
      TT=(WL+SS*TAIL)*TAIL
      VT=(RT**0.666666)*(S**0.5)/MANER
      QT=TT*VT
      HI=QT-QPEAK
      TAIL=TAIL+FT
      XT=XT-1
      IF(XT.EQ.0.) GO TO 8
      HT1=HI
      GO TO 7
8  HI=(HI-HT1)/FT
      HT1=G-HT1/HI

```

```

IF(ABS(HT1-G).LT.ERR)GO TO 9
TAIL=HT1
WRITE(*,51)TAIL
51 FORMAT(10X,'TRIAL DEPTH =',F6.2)
GO TO 6
9 IF(VT.LT.VELC)GO TO 10
WL=WL+WL/4
GO TO 11
10 TAIL=G
WRITE(*,50)WL,TAIL,SS,VT
WRITE(15,50)WL,TAIL,SS,VT
50 FORMAT(10X,'* TYPICAL SECTION AT DOWNSTREAM *',/,15X,'WIDTH (m) ='
1,F6.2,/,15X,'DEPTH OF FLOW (m) =' ,F6.2,/,15X,'SIZE SLOPE 1:' ,F4.2
2,/,15X,'ALLOWED VELOCITY AT DOWNSTREAM (m) =' ,F6.2)
C DESIGN CHUTE SECTION
FCH=FT
QCR=QPEAK/BCH
DCR=((QCR**2)/9.807)**0.33333)
VEL(1)=QPEAK/BCH/DCR
DEPTH(1)=DCR
HV(1)=((VEL(1))**2)/19.62
R1=(BCH*DCR)/(BCH+2*DCR)
SF1=((VEL(1)*MANCH)/(R1**0.66666))**2.
SL=ELV(1)+DCR+HV(1)
DO 12 I=2,NOSTA
DEPTH(I)=DEPTH(I-1)/2
GO TO 13
57 DEPTH(I)=DEPTH(I-1)/3
13 GCH=DEPTH(I)
XCH=2.
14 IF(DEPTH(I).LT.0)GO TO 57
R2=(BCH*DEPTH(I))/(BCH+2*DEPTH(I))
WRITE(*,56)I,R2,DEPTH(I)
56 FORMAT(10X,'STA NO=',I2,5X,'R2=',F6.2,10X,'DEPTH2=',F6.2)
VEL(I)=QCR/DEPTH(I)
HV(I)=((VEL(I))**2)/19.62
SF2=((VEL(I)*MANCH)/(R2**0.66666))**2
HLOSS=(SF1+SF2)*(STA(I)-STA(I-1))/2
RS=ELV(I)+DEPTH(I)+HV(I)+HLOSS
HICH=RS-SL
DEPTH(I)=DEPTH(I)+FCH
XCH=XCH-1
IF(XCH.EQ.0.)GO TO 15
HCH=HICH
GO TO 14
15 HICH=(HICH-HCH)/FCH
HCH=GCH-HCH/HICH
IF(ABS(HCH-GCH).LT.ERR)GO TO 16
DEPTH(I)=HCH
GO TO 13
16 DEPTH(I)=GCH
SL=ELV(I)+DEPTH(I)+HV(I)
SF1=SF2
12 CONTINUE
D1=DEPTH(NOSTA)

```

```

V1=VEL(NOSTA)
FRNO=V1/((9.807*D1)**0.5)
D2=D1*((1+8*(FRNO**2))**0.5-1)/2
WRITE(*,19)BCH
WRITE(15,19)BCH
19 FORMAT(10X,'>>>>DESIGN CHUTE SPILLWAY<<<<',/,10X,'OPENNING SPILLWA
1Y (m) =',F10.2,/, 'STATION (m)',5X,'ELEVATION (m)',5X,'DEPTH OF FLO
2W (m)',/)
WRITE(*,72)(STA(I),ELV(I),DEPTH(I),I=1,NOSTA)
WRITE(15,72)(STA(I),ELV(I),DEPTH(I),I=1,NOSTA)
72 FORMAT(10X,F6.2,6X,F6.2,6X,F6.2)
C DESIGN VERTICAL CURVE
NOSTA=NOSTA-1
DO 17 I=1,NOSTA
IF((SLOPE(I)-SLOPE(I+1)).LT.0.)GO TO 21
WRITE(*,18)STA(I)
WRITE(15,18)STA(I)
RADUIS=2*DEPTH(I)*((3.28*VEL(I))**2)/100
WRITE(*,20)RADUIS
WRITE(15,20)RADUIS
20 FORMAT(10X,'SAG CURVE =====> WARNING MINIMUM RADUIS (m)= ',F10.2)
GO TO 17
21 SETA=ATAN(SLOPE(I+1))
LO=(SLOPE(I+1)-SLOPE(I))*2*(DEPTH(I)+HV(I))*(COS(SETA))**2/0.5
NO=LO+1
WRITE(*,18)STA(I)
WRITE(15,18)STA(I)
18 FORMAT(10X,'CURVE AT STA =',F10.2)
WRITE(*,23)
WRITE(15,23)
23 FORMAT(10X,'CREST CURVE',/,10X,'COORDINATE CURVE',/,
1'DISTANCE X (m)',18X,'DEPTH Y (m)',/)
DO 22 J=1,NO
Y11=J*SLOPE(J)+(SLOPE(J+1)-SLOPE(J))*(J**2)/2/NO
WRITE(*,24)J,Y11
WRITE(15,24)J,Y11
24 FORMAT(10X,I2,18X,F10.2)
22 CONTINUE
17 CONTINUE
C DESIGN STILLING BASIN
WALL=ELV(NOSTA+1)+0.1*(V1+D2)+D2
IF(FRNO.LT.4.5)GO TO 29
WRITE(*,35)
WRITE(15,35)
35 FORMAT(10X,'>>>> DESIGN STILLING BASIN USBR TYPE III <<<<',/)
DO 25 I=1,12
IF(FRNO.GT.FR3(I))GO TO 25
DL=((FRNO-FR3(I-1))/(FR3(I)-FR3(I-1)))*(DL3(I)-DL3(I-1))+DL3(I-1)
BASIN=DL*D2
BAFF=0.8*D2
H4=(0.0571428*FRNO+1.2)*D1
H3=(0.1785714*FRNO+1.25)*D1
SPACE=0.5*D1
WRITE(*,26)D1,D1,D1,SPACE
WRITE(15,26)D1,D1,D1,SPACE

```

```

26 FORMAT(10X,'* CHUTE BLOCK *',/,15X,'HEIGHT (m) =',F6.2,/,15X,
1'WIDTH (m) =',F6.2,/,15X,'SPACING BETWEEN CHUTE BLOCK (m) =',F6.2
2,/,15X,'SPACING BETWEEN CHUTE BLOCK TO RETAINING WALL (m) =',F6.2)
WIDTH=0.75*H3
SPACE1=0.2*H3
SPACE2=0.375*H3
WRITE(*,27)H3,WIDTH,WIDTH,SPACE1,SPACE2
WRITE(15,27)H3,WIDTH,WIDTH,SPACE1,SPACE2
27 FORMAT(10X,'* BAFFLE BLOCK *',/,15X,'HEIGHT (m) =',F6.2,/,15X,
1'WIDTH (m) =',F6.2,/,15X,'SPACING BETWEEN BAFFLE BLOCK (m) =',F6.2
2,/,15X,'THICKNESS (m) =',F6.2,/,15X,'SPACING BAFFLE BLOCK TO RETAI
3NING WALL (m) =',F6.2,)
WRITE(*,28)H4,BASIN,BAFF
WRITE(15,28)H4,BASIN,BAFF
28 FORMAT(10X,'* END SILL *',/,15X,'HEIGHT (m) =',F6.2,/,10X,'LENGTH
1OF STILLING BASIN (m) =',F6.2,/,10X,'LENGTH OF BAFFLE BLOCK TO CH
2UTE BLOCK (m) =',F6.2)
GO TO 34
25 CONTINUE
29 WRITE(*,30)
WRITE(15,30)
30 FORMAT(10X,'>>>> DESIGN STILLING BASIN TYPE IV <<<<',/)
DO 31 I=1,4
IF(FRNO.GT.FR4(I))GO TO 31
DL=((FRNO-FR4(I-1))/(FR4(I)-FR4(I-1)))*(DL4(I)-DL4(I-1))+DL4(I-1)
BASIN=DL*D2
HEIGHT=2.0*D1
WIDTH=D1
ENGTH=2.5*D1
SPACE=1.25*D1
WRITE(*,33)HEIGHT,WIDTH,ENGTH,WIDTH,SPACE,BASIN
WRITE(15,33)HEIGHT,WIDTH,ENGTH,WIDTH,SPACE,BASIN
33 FORMAT(10X,'* CHUTE BLOCK *',/,15X,'HEIGHT (m) =',F6.2,/,15X,'WIDT
1H (m) =',F6.2,/,15X,'LENGTH (m) =',F6.2,/,15X,'SPACING BETWEEN CHU
2TE BLOCK (m) =',F6.2,/,10X,'* END SILL *',/,15X,'HEIGHT (m) =',
3F6.2,/,15X,'LENGTH OF STILLING BASIN (m) =',F6.2)
GO TO 34
31 CONTINUE
34 WRITE(*,36)WALL
WRITE(15,36)WALL
36 FORMAT(10X,'HEIGHT OF RETAINING WALL (m) =',F6.2)
C
C   CREATED GRAPHIC DRAWING FILE TRANSFER TO AUTOCAD
C
NOSTA=NOSTA+1
WALL=D2+0.1*(V1+D2)
X(1)=100.
Y(1)=ELV(1)-ELV(NOSTA)+110.
X(2)=112.
Y(2)=Y(1)
DO 38 I=2,NOSTA
X(I+1)=(STA(I)-STA(I-1))+X(I)
Y(I+1)=Y(I)-(ELV(I-1)-ELV(I))
38 CONTINUE
X(NOSTA+2)=X(NOSTA+1)+BASIN

```

```

Y(NOSTA+2)=Y(NOSTA+1)
X(NOSTA+3)=X(NOSTA+2)
Y(NOSTA+3)=Y(NOSTA+1)+WALL
X(NOSTA+4)=X(NOSTA+1)
Y(NOSTA+4)=Y(NOSTA+3)
X(NOSTA+5)=X(NOSTA)
Y(NOSTA+5)=Y(NOSTA)+1.00
WRITE(21,39)
39 FORMAT('LINE')
DO 43 JJ=1,NOSTA+5
WRITE(21,61)X(JJ),Y(JJ)
61 FORMAT(F6.2,',',F6.2)
43 CONTINUE
X(NOSTA+6)=X(1)
Y(NOSTA+6)=Y(1)
X(NOSTA+7)=X(1)
Y(NOSTA+7)=Y(1)+1.50
X(NOSTA+8)=X(1)+3.00
Y(NOSTA+8)=Y(1)+2.50
X(NOSTA+9)=X(NOSTA+8)+6.00
Y(NOSTA+9)=Y(1)+2.50
X(NOSTA+10)=X(NOSTA+9)+3.00
Y(NOSTA+10)=Y(1)+1.50
DO 45 I=1,NOSTA-2
X(NOSTA+10+I)=X(I+2)
Y(NOSTA+10+I)=Y(I+2)+1.00
45 CONTINUE
N11=NOSTA+5
NN=NOSTA+3
WRITE(21,419)X(N11+1),Y(N11+1)
DO 47 JJ=1,NN
WRITE(21,41)X(JJ+N11),Y(JJ+N11)
41 FORMAT(F6.2,',',F6.2)
47 CONTINUE
N1=2*NOSTA+8
X(N1+1)=X(1)
Y(N1+1)=Y(1)
X(N1+2)=X(1)
Y(N1+2)=Y(1)-0.2
DO 411 I=1,NOSTA-1
X(N1+2+I)=X(I+1)
Y(N1+2+I)=Y(I+1)-0.2
411 CONTINUE
NO=N1+NOSTA+1
X(NO+1)=X(NOSTA+1)
Y(NO+1)=Y(NOSTA+1)-0.90
X(NO+2)=X(NOSTA+2)
Y(NO+2)=Y(NOSTA+2)-0.40
C DRAW END SILL
X(NO+3)=X(NOSTA+2)
Y(NO+3)=Y(NOSTA+2)+H4
X(NO+4)=X(NOSTA+2)-0.10
Y(NO+4)=Y(NO+3)
X(NO+5)=X(NOSTA+2)-0.10-(H4)*2.
Y(NO+5)=Y(NOSTA+2)

```

```

      NN=NO+5-N1
      WRITE(21,419)X(N1+1),Y(N1+1)
      DO 418 JJ=1,NN
      WRITE(21,61)X(N1+JJ),Y(N1+JJ)
418 CONTINUE
C   DRAW BAFFLE BLOCK
      N1=NO+5
      X(NO+6)=X(NOSTA+1)+BAFF
      Y(NO+6)=Y(NOSTA+2)
      X(NO+7)=X(NO+6)
      Y(NO+7)=Y(NO+6)+H3
      X(NO+8)=X(NO+7)+SPACE1
      Y(NO+8)=Y(NO+7)
      X(NO+9)=X(NO+8)+H3
      Y(NO+9)=Y(NOSTA+2)
      WRITE(21,419)X(N1+1),Y(N1+1)
419 FORMAT(/,'LINE',1X,F6.2,',',F6.2)
      DO 412 JJ=1,4
      WRITE(21,61)X(JJ+N1),Y(JJ+N1)
412 CONTINUE
C   DRAW CHUTE BLOCK
      N12=NO+9
      X(NO+10)=X(NOSTA+1)
      Y(NO+10)=Y(NOSTA+1)
      X(NO+11)=X(NOSTA+1)
      Y(NO+11)=Y(NOSTA+1)+D1
      X(NO+12)=X(NOSTA+1)-2*D1
      Y(NO+12)=Y(NO+11)
      WRITE(21,419)X(N12+1),Y(N12+1)
      DO 417 JJ=1,3
      WRITE(21,61)X(N12+JJ),Y(N12+JJ)
417 CONTINUE
C   DRAW CUTOFF UPSTREAM
      N12=NO+12
      X(NO+13)=X(1)
      Y(NO+13)=Y(1)-0.20
      X(NO+14)=X(1)
      Y(NO+14)=Y(NO+13)-1.80
      X(NO+15)=X(1)+0.20
      Y(NO+15)=Y(NO+14)
      X(NO+16)=X(NO+15)
      Y(NO+16)=Y(NO+13)
      WRITE(21,419)X(N12+1),Y(N12+1)
      DO 414 JJ=1,4
      WRITE(21,61)X(JJ+N12),Y(JJ+N12)
414 CONTINUE
C   DRAW CUTOFF EVERY SECTION OF CHUTE
      J1=0
      DO 423 I=1,NOSTA
      X(NO+17+J1)=X(I+1)
      Y(NO+17+J1)=Y(I+1)-0.20
      X(NO+18+J1)=X(I+1)
      Y(NO+18+J1)=Y(I+1)-1.80
      X(NO+19+J1)=X(I+1)+0.20
      Y(NO+19+J1)=Y(I+1)-1.80

```

```

      X(NO+20+J1)=X(I+1)+0.20
      Y(NO+20+J1)=Y(I+1)-0.20
      N13=NO+16+J1
      WRITE(21,419)X(N13+1),Y(N13+1)
      DO 415 JJ=1,4
      WRITE(21,61)X(JJ+N13),Y(JJ+N13)
415  CONTINUE
      J1=J1+4
423  CONTINUE
C    DRAW CUTOFF AT ENDSILL
      NO1=JJ+N13
      X(NO1+1)=X(NOSTA+2)
      Y(NO1+1)=Y(NO+2)
      X(NO1+2)=X(NO1+1)
      Y(NO1+2)=Y(NO1+1)-1.60
      X(NO1+3)=X(NO1+2)-0.20
      Y(NO1+3)=Y(NO1+2)
      X(NO1+4)=X(NO1+3)
      Y(NO1+4)=Y(NO1+1)
      WRITE(21,419)X(NO1+1),Y(NO1+1)
      DO 416 JJ=1,4
      WRITE(21,61)X(NO1+JJ),Y(NO1+JJ)
416  CONTINUE
      WRITE(21,61)X(NO1+4),Y(NO1+4)
C
C    PLAN SPILLWAY
C    OPENNING SPILLWAY=WL
C
      X1(1)=100.00
      Y1(1)=100.00
      X1(2)=X(NOSTA+2)
      Y1(2)=Y1(1)
      X1(3)=X1(2)
      Y1(3)=Y1(2)+0.15
      X1(4)=X1(1)
      Y1(4)=Y1(3)
      X1(5)=X1(1)
      Y1(5)=Y1(1)
      WRITE(22,621)X1(1),Y1(1)
621  FORMAT('LINE',1X,F6.2,',',',',F6.2)
      DO 62 JJ=1,4
      WRITE(22,61)X1(JJ),Y1(JJ)
62  CONTINUE
      X1(6)=X1(5)
      Y1(6)=Y1(5)
      X1(7)=X1(6)
      Y1(7)=Y1(1)+WL/2
      WRITE(22,622)X1(6),Y1(6)
622  FORMAT(/,'LINE',1X,F6.2,',',',',F6.2)
      DO 626 JJ=1,2
      WRITE(22,61)X1(5+JJ),Y1(5+JJ)
626  CONTINUE
      J1=0
      DO 53 I=1,NOSTA
      X1(J1+8)=X(I+1)

```

```

Y1(J1+8)=Y1(1)
X1(J1+9)=X(I+1)
Y1(J1+9)=Y1(7)
WRITE(22,622)X1(J1+8),Y1(J1+8)
DO 627 JJ=1,2
WRITE(22,61)X1(J1+7+JJ),Y1(J1+7+JJ)
627 CONTINUE
J1=J1+2
53 CONTINUE
C DRAW ENDSILL
X1(2*NOSTA+8)=X(NOSTA+2)
Y1(2*NOSTA+8)=Y1(1)
NO3=2*NOSTA+9
X1(NO3)=X(NOSTA+2)
Y1(NO3)=Y1(7)
WRITE(22,622)X1(NO3-1),Y1(NO3-1)
DO 628 JJ=1,2
WRITE(22,61)X1(NO3-2+JJ),Y1(NO3-2+JJ)
628 CONTINUE
C
X1(NO3+1)=X1(4)
Y1(NO3+1)=Y1(4)+1.50
X1(NO3+2)=X1(NO3)
Y1(NO3+2)=Y1(NO3+1)
WRITE(22,622)X1(NO3+1),Y1(NO3+1)
DO 629 JJ=1,2
WRITE(22,61)X1(NO3+JJ),Y1(NO3+JJ)
629 CONTINUE
C
X1(NO3+3)=X1(NO3)-0.10
Y1(NO3+3)=Y1(1)+0.15
X1(NO3+4)=X1(NO3+3)
Y1(NO3+4)=Y1(7)
WRITE(22,622)X1(NO3+3),Y1(NO3+3)
DO 630 JJ=1,2
WRITE(22,61)X1(NO3+2+JJ),Y1(NO3+2+JJ)
630 CONTINUE
C
X1(NO3+5)=X1(NO3+4)-H4*2
Y1(NO3+5)=Y1(NO3+3)
X1(NO3+6)=X1(NO3+5)
Y1(NO3+6)=Y1(7)
WRITE(22,622)X1(NO3+5),Y1(NO3+5)
DO 631 JJ=1,2
WRITE(22,61)X1(NO3+4+JJ),Y1(NO3+4+JJ)
631 CONTINUE
C
X1(NO3+7)=X(NOSTA+1)
Y1(NO3+7)=Y1(NO3+3)+SPACE
X1(NO3+8)=X1(NO3+7)-D1*2
Y1(NO3+8)=Y1(NO3+7)
X1(NO3+9)=X1(NO3+8)
Y1(NO3+9)=Y1(NO3+8)+D1
X1(NO3+10)=X1(NO3+7)
Y1(NO3+10)=Y1(NO3+9)

```

```

WRITE(22,419)X1(NO3+7),Y1(NO3+7)
DO 623 JJ=1,4
WRITE(22,61)X1(NO3+6+JJ),Y1(NO3+6+JJ)
623 CONTINUE
C   DRAW BAFFLE BLOCK
X1(NO3+11)=X(NOSTA+1)+BAFF+SPACE1+H3
Y1(NO3+11)=Y1(NO3+3)+SPACE2
X1(NO3+12)=X(NOSTA+1)+BAFF
Y1(NO3+12)=Y1(NO3+11)
X1(NO3+13)=X1(NO3+12)
Y1(NO3+13)=Y1(NO3+12)+WIDTH
X1(NO3+14)=X1(NO3+11)
Y1(NO3+14)=Y1(NO3+13)
X1(NO3+15)=X1(NO3+14)
Y1(NO3+15)=Y1(NO3+12)
WRITE(22,419)X1(NO3+11),Y1(NO3+11)
DO 624 JJ=1,5
WRITE(22,61)X1(NO3+JJ+10),Y1(NO3+JJ+10)
624 CONTINUE
C   DRAW THICK BAFFLE BLOCK
X1(NO3+16)=X1(NO3+12)+SPACE1
Y1(NO3+16)=Y1(NO3+11)
X1(NO3+17)=X1(NO3+16)
Y1(NO3+17)=Y1(NO3+16)+WIDTH
WRITE(22,622)X1(NO3+16),Y1(NO3+16)
DO 632 JJ=1,2
WRITE(22,61)X1(NO3+15+JJ),Y1(NO3+15+JJ)
632 CONTINUE
WRITE(22,61)X1(NO3+17),Y1(NO3+17)
C
C   DOWN STREAM SECTION
C
X2(1)=12.50
Y2(1)=20.00
X2(2)=10.00
Y2(2)=20.00
X2(3)=10.00
Y2(3)=20.30
X2(4)=X2(3)+2.25
Y2(4)=Y2(3)
X2(5)=X2(4)+0.10
Y2(5)=Y2(4)+WALL
X2(6)=X2(5)+0.15
Y2(6)=Y2(5)
X2(7)=X2(6)
Y2(7)=Y2(4)-2.00
X2(8)=X2(7)+WL/2
Y2(8)=Y2(7)
WRITE(23,591)X2(1),Y2(1)
DO 362 JJ=1,8
WRITE(23,691)X2(JJ),Y2(JJ)
362 CONTINUE
C
X2(9)=X2(8)
Y2(9)=Y2(1)-(0.50*BAFF/BASIN)

```

```

      X2(10)=X2(7)
      Y2(10)=Y2(9)
      WRITE(23,692)X2(9),Y2(9)
      DO 633 JJ=1,2
      WRITE(23,691)X2(8+JJ),Y2(8+JJ)
633  CONTINUE
C
      X2(11)=X2(1)
      Y2(11)=Y2(3)
      X2(12)=X2(8)
      Y2(12)=Y2(3)
      WRITE(23,692)X2(11),Y2(11)
      DO 634 JJ=1,2
      WRITE(23,691)X2(10+JJ),Y2(10+JJ)
634  CONTINUE
C
      X2(13)=X2(7)
      Y2(13)=Y2(3)
      X2(14)=X2(7)+SPACE2
      Y2(14)=Y2(3)
      X2(15)=X2(14)
      Y2(15)=Y2(14)+H3
      X2(16)=X2(15)+WIDTH
      Y2(16)=Y2(15)
      X2(17)=X2(16)
      Y2(17)=Y2(3)
      X2(18)=X2(17)+WIDTH
      Y2(18)=Y2(3)
      WRITE(23,692)X2(13),Y2(13)
      DO 635 JJ=1,6
      WRITE(23,691)X2(12+JJ),Y2(12+JJ)
635  CONTINUE
      WRITE(23,691)X2(18),Y2(18)
C
C      SECTION UP STREAM
C
      X3(1)=14.00
      Y3(1)=20.30
      X3(2)=X3(1)
      Y3(2)=20.00
      X3(3)=10.00
      Y3(3)=20.00
      X3(4)=10.00
      Y3(4)=20.30
      X3(5)=12.25
      Y3(5)=20.30
      X3(6)=12.35
      Y3(6)=21.80
      X3(7)=12.50
      Y3(7)=21.80
      X3(8)=12.50
      Y3(8)=20.30
      X3(9)=14.00
      Y3(9)=20.30
      WRITE(24,591)X3(1),Y3(1)

```

```

591 FORMAT('LINE',1X,F5.2,',',F5.2)
    DO 636 JJ=1,9
      WRITE(24,691)X3(JJ),Y3(JJ)
636 CONTINUE
691 FORMAT(F5.2,',',F5.2)
C
    X3(10)=X3(9)
    Y3(10)=Y3(9)
    X3(11)=X3(10)+(WL/2-1.50)
    Y3(11)=Y3(9)
    WRITE(24,692)X3(10),Y3(10)
692 FORMAT(/,'LINE',1X,F5.2,',',F5.2)
    DO 637 JJ=1,2
      WRITE(24,691)X3(9+JJ),Y3(9+JJ)
637 CONTINUE
C
    X3(12)=X3(10)
    Y3(12)=Y3(10)-0.20
    X3(13)=X3(11)
    Y3(13)=Y3(12)
    WRITE(24,692)X3(12),Y3(12)
    DO 638 JJ=1,2
      WRITE(24,691)X3(11+JJ),Y3(11+JJ)
638 CONTINUE
    WRITE(24,691)X3(13),Y3(13)
C
C   SECTION C - C
C
    X4(1)=12.50
    Y4(1)=20.00
    X4(2)=10.00
    Y4(2)=20.00
    X4(3)=X4(2)
    Y4(3)=20.30
    X4(4)=X4(3)+2.25
    Y4(4)=20.30
    X4(5)=X4(4)+0.10
    Y4(5)=20.30+WALL
    X4(6)=X4(5)+0.15
    Y4(6)=Y4(5)
    X4(7)=X4(6)
    Y4(7)=Y4(1)-2.00
    X4(8)=X4(7)+WL/2
    Y4(8)=Y4(7)
    X4(9)=X4(8)
    Y4(9)=Y4(3)+H4
    X4(10)=X4(1)
    Y4(10)=Y4(3)+H4
    WRITE(25,591)X4(1),Y4(1)
    DO 639 JJ=1,10
      WRITE(25,691)X4(JJ),Y4(JJ)
639 CONTINUE
    X4(11)=X4(9)
    Y4(11)=20.30
    X4(12)=X4(9)+WL/2

```

```

Y4(12)=20.30
X4(13)=X4(12)
Y4(13)=20.30+WALL
X4(14)=X4(12)+0.15
Y4(14)=Y4(13)
X4(15)=X4(12)+0.25
Y4(15)=Y4(12)
X4(16)=X4(15)+2.25
Y4(16)=Y4(12)
X4(17)=X4(16)
Y4(17)=Y4(12)-0.30
X4(18)=X4(17)-2.50
Y4(18)=Y4(17)
X4(19)=X4(18)
Y4(19)=Y4(18)-(0.50*BAFF/BASIN)
X4(20)=X4(9)
Y4(20)=Y4(19)
WRITE(25,692)X4(11),Y4(11)
DO 640 JJ=1,10
WRITE(25,691)X4(10+JJ),Y4(10+JJ)

```

640 CONTINUE

C

```

X4(21)=X4(12)-SPACE
Y4(21)=Y4(12)
X4(22)=X4(21)
Y4(22)=Y4(21)+D1
X4(23)=X4(22)-D1
Y4(23)=Y4(22)
X4(24)=X4(23)
Y4(24)=Y4(21)
WRITE(25,692)X4(21),Y4(21)
DO 641 JJ=1,4
WRITE(25,691)X4(20+JJ),Y4(20+JJ)

```

641 CONTINUE

C

```

X4(25)=X4(12)
Y4(25)=Y4(12)
X4(26)=X4(25)
Y4(26)=Y4(17)
WRITE(25,692)X4(25),Y4(25)
DO 642 JJ=1,2
WRITE(25,691)X4(24+JJ),Y4(24+JJ)

```

642 CONTINUE

```

WRITE(25,691)X4(26),Y4(26)

```

C

C

* DAM SECTION * GRAPHIC

C

```

READ(19,75)NODAM,ELVA,ELVC,GL,Z1,Z2
75 FORMAT(I8,5F8.2)
READ(19,77)(ELVB(I),I=1,NODAM)
77 FORMAT(F8.2)
DO 79 I=1,NODAM
WRITE(26,852)I
852 FORMAT(1X,'CROSS SECTION DAM STA NO. ',I3)
X5(1)=100.00

```

```

Y5(1)=120.00
X5(2)=115.00
Y5(2)=120.00
X5(3)=(ELVA-GL)*Z1+X5(2)
Y5(3)=(ELVA-GL)+Y5(1)
X5(4)=X5(3)+6.00
Y5(4)=Y5(3)
X5(5)=X5(4)+(ELVA-GL)*Z2
Y5(5)=Y5(2)
X5(6)=X5(5)-1.70
Y5(6)=Y5(5)-1.70
X5(7)=X5(6)-1.20
Y5(7)=Y5(6)
X5(8)=X5(7)-1.20
Y5(8)=Y5(5)-0.50

```

C

```

X5(10)=X5(3)+5
Y5(10)=Y5(5)-(GL-ELVB(I))
X5(11)=X5(10)-4.00
Y5(11)=Y5(10)
X5(12)=X5(11)-(Y5(8)-Y5(11))
Y5(12)=Y5(8)
X5(13)=X5(2)+0.50
Y5(13)=Y5(2)-0.50
X5(9)=X5(10)+(GL-0.5-ELVB(I))
Y5(9)=Y5(8)
X5(14)=X5(2)
Y5(14)=Y5(2)

```

C

```

WRITE(26,621)X5(1),Y5(1)
DO 643 JJ=1,14
WRITE(26,61)X5(JJ),Y5(JJ)

```

643 CONTINUE

C

```

79 CONTINUE
CLOSE(5)
CLOSE(9)
CLOSE(15)
CLOSE(19)
CLOSE(21)
CLOSE(22)
CLOSE(23)
CLOSE(24)
CLOSE(25)
CLOSE(26)
RETURN
END

```

\$DEBUG

SUBROUTINE UNITH

C

C

C

```

FLOOD PEAK BY SYSTHETIC UNIT HYDROGRAPH
MODIFIED BY SCS (SOIL CONSERVATION SERVICE)
DIMENSION H(440)
DIMENSION UH(150),PERCNT(150),R(288),RAIN(288)
DIMENSION DYT(48)

```

```

C
C   ADJUST POINT RAIN FALL
C
C   DATA DYR/0.,20.,40.,60.,80.,100.,125.,150.,100.,89.4,82.3,78.,
C74.8,72.4,70.2,68.5,100.,94.8,91.2,88.4,86.3,84.5,82.7,81.3,
C100.,96.2,93.8,91.8,90.3,89.,87.8,86.7,100.,97.4,95.8,94.8,
C93.9,93.2,92.8,92.3,100.,82.,72.5,66.6,63.,60.8,59.,57.5/
C   DATA UH/150*0./
C   DATA PERCNT/150*0./
C   DATA R/288*0./
C   DATA H/440*0./
C
C   FUNCTION
C
C   RR(T)=EXP(A*ALOG(T)+B)
C
C   READ DATA INPUT
C   KTYPE = Select 24-hour storm unit interval model number
C       1 = Duration 5-minutes           2 = Duration 10-minutes
C       3 = Duration 15-minutes          4 = Duration 20-minutes
C       5 = Duration 30-minutes          6 = Duration 60-minutes
C   KODE1 = Select appropriate S-Graph
C       1 = Valley zone                 2 = Foothill zone   3=Mountain zone
C       4 = Desert zone                 5 = Combination of options 1-4
C       6 = U.S. soil conservation service (SCS)
C   KCODE2 = TO select SI unit or EI unit ....if use SI KODE2=1
C   HH = Difference in elevation between the most remote point
C       in the watershed and the point of concentration ...m
C
C   OPEN(3,FILE='UNITH.PRN')
C   OPEN(8,FILE='ROUT.DAT')
C   OPEN(13,FILE='UNITH.RES')
C   READ(3,10)KTYPE,KODE1,KCODE2,HH
C   WRITE(*,10)KTYPE,KODE1,KCODE2,HH
10  FORMAT(//////////////////////////////////////3I8,F8.2)
C   CONVERT M TO FT
C
C   XL = the length of the longest watercourse.....km
C   XLCA= the length along the longest watershed,Watercourse measure
C       from the point of concentration upstream to a point opposite
C       the centroid of the watershed area.....km
C   XN = Basin factor (Ref.U.S.crop of Engineer)
C   AREA= watershed area ( km^2 )
C   VSL = adjust loss rate (cm/hr)
C   BASCON=baseflow (cms/square km)
C   SLP = low loss rate percentage (decimal notation)
C
C   READ(3,20)XL,XLCA,XN,AREA,VSL,BASCON,SLP
C   WRITE(*,20)XL,XLCA,XN,AREA,VSL,BASCON,SLP
20  FORMAT(7F8.2)
C
C   RAINFALL RETURN PERIOD FOR 25-YEARS
C       watershed area averaged point rainfall    depth-area adj.factor
C       5-minute ["R5 "]                        ["FX5 "]
C       30-minute ["R30"]                       ["FX30"]

```

```

C          1-hr.      ["R1 "]          ["FX1 "]
C          3-hr.      ["R3 "]          ["FX3 "]
C          6-hr.      ["R6 "]          ["FX6 "]
C          24-hr.     ["R24"]          ["FX24"]
C
      READ(3,30)R5,R30,R1,R3,R6,R24
      WRITE(*,30)R5,R30,R1,R3,R6,R24
30  FORMAT(6F8.2)
C
C      EFFECTIVE RAINFALL INFORMATION DISPLAY OPTIONS:
C          0 = omit effective rainfall information from results
C          1 = include effective rainfall information with results
C          KSOIL = select display option desired
C          KSTORM=
C
      READ(3,40)KSOIL
      WRITE(*,40)KSOIL
40  FORMAT(18)
C
C      PERCENTAGE [DECIMAL NOTATION] OF WATERSHED SPECIFIED
C          Valley  " S " Curve.....=> " PV "
C          Foothill " S " Curve.....=> " PF "
C          Mountain " S " Curve.....=> " PM "
C          Desert  " S " Curve.....=> " PD "
C
      READ(3,50)PV,PF,PM,PD
      WRITE(*,50)PV,PF,PM,PD
50  FORMAT(4F8.2)
C
C      DEPTH-AREA ADJUSTMENT FACTOR
C          5-minute      [ "FX5 " ]
C          30-minute     [ "FX30" ]
C          1-hour        [ "FX1 " ]
C          3-hour        [ "FX3 " ]
C          6-hour        [ "FX6 " ]
C          24-hour       [ "FX24" ]
C
      READ(3,60)FX5,FX30,FX1,FX3,FX6,FX24
      WRITE(*,60)FX5,FX30,FX1,FX3,FX6,FX24
60  FORMAT(6F8.2)
C
C      PRECIPITATION DEPTH-AREA FACTOR MODELS:
C          DAOPT = Select desired model
C              = 1:NOAA atlas II values (30-minute value used for 5-min
C                depth)
C              = 2:User-specified depth-area factors
C          TIME1 = time(hr.) for beginning of results
C          TIME2 = time(hr.) for end of results
C
      READ(3,70)DAOPT,TIME1,TIME2
      WRITE(*,70)DAOPT,TIME1,TIME2
70  FORMAT(F8.0,2F8.2)
      IF(KCODE2.NE.1)GO TO 300
      HH=HH*3.28
C      CONVERT km  TO ft                      (XL) & (XLCA)

```

```

C          km^2 = acres          (AREA)
C          cm/hr=inches/hr      (VSL)
C          cms/hr/square km=cfs/acres(BASCON)
XL=XL*3280.
XLCA=XLCA*3280.
AREA=AREA*255.
VSL=VSL*.3936
BASCON=BASCON*0.13838
C  CONVERT cm TO inch
R5=R5*0.3936
R30=R30*0.3936
R1=R1*0.3936
R3=R3*0.3936
R6=R6*0.3936
R24=R24*0.3936
300 AX=AREA
    XLX=XL
    XLCAX=XLCA
C
    FX=1.
C
    IF(DAOPT.EQ.2.)GO TO 183
C
    AREA=AREA/640.
    FX1=.651
    FX3=.78
    FX6=.831
    FX24=.91
    IF(AREA.GE.350.)GO TO 180
C
    DO 100 I=1,13
    IF(AREA.LE.DYR(I))GO TO 110
100 CONTINUE
C
110 I=I-1
    DX=DYR(I+1)-DYR(I)
    FACT=AREA-DYR(I)
    FX1=(DYR(8+I)-FACT*(DYR(8+I)-DYR(9+I))/DX)/100.
    FX3=(DYR(16+I)-FACT*(DYR(16+I)-DYR(17+I))/DX)/100.
    FX6=(DYR(24+I)-FACT*(DYR(24+I)-DYR(25+I))/DX)/100.
    FX24=(DYR(32+I)-FACT*(DYR(32+I)-DYR(33+I))/DX)/100.
    FX30=(DYR(40+I)-FACT*(DYR(40+I)-DYR(41+I))/DX)/100.
    FX5=FX30
180 CONTINUE
    AREA=AREA*640.
C
C
183 CONTINUE
    WRITE(*,5000)
    WRITE(13,5000)
    WRITE(*,181)
    WRITE(13,181)
181 FORMAT(/,14X,'UNIT - HYDROGRAPH ANALYSIS',
. /)
    WRITE(*,5000)

```

```

        WRITE(13,5000)
6666 FORMAT(/,10X,'*USER SPECIFIED PRECIPITATION DEPTH-AREA ',
        . 'REDUCTION FACTORS:')
6665 FORMAT(/,10X,'PRECIPITATION DEPTH-AREA REDUCTION FACTORS:')
6664 FORMAT(
        .12X,'5-MINUTE FACTOR =',F5.3,/,12X,'30-MINUTE FACTOR =',
        .F5.3,/,
        C12X,'1-HOUR FACTOR =',F5.3,/,12X,'3-HOUR FACTOR =',F5.3,/,
        C12X,'6-HOUR FACTOR =',F5.3,/,12X,'24-HOUR FACTOR =',F5.3)
C
C DETERMINE HYDROGRAPH FACTORS
C
        XL=XL/5280.
        XLCA=XLCA/5280.
        S=HH/XLX
        XLAGX=1.2*(XL*XLCA/S**0.5)**0.38
        XLAG=20.0*XLX*XLAGX
C
C DESIGNATE UNIT INTERVALS
C
        IF(KTYPE.EQ.1)UNIT=5.
        IF(KTYPE.EQ.2)UNIT=10.
        IF(KTYPE.EQ.3)UNIT=15.
        IF(KTYPE.EQ.4)UNIT=20.
        IF(KTYPE.EQ.5)UNIT=30.
        IF(KTYPE.EQ.6)UNIT=60.
C
        TIMLAG=100.*UNIT/60./XLAG
        SQMI=AREA*43560./5280./5280.
        BASFLO=BASCON*SQMI
        XK=645.*SQMI*60./UNIT
        XK=XK/100.
C
C RETURN XL AND XLCA TO UNITS OF FEET
C
        XL=XL*5280.
        XLCA=XLCA*5280.
5000 FORMAT(1X,76('*'))
C
        WRITE(*,5000)
        WRITE(13,5000)
        IF(KCODE2.NE.1)GO TO 7001
        XLX1=XLX/3.28
        XLCAX1=XLCAX/3.28
        HH1=HH/3.28
        AX1=AX*0.00392
        BASCON1=BASCON*0.010936
        VSL1=VSL*2.54
        WRITE(*,5001)XLX,XLCAX,HH,XN,AX,XLAG,UNIT,TIMLAG,
        CBASCON,VSL,SLP
        WRITE(13,7000)XLX1,XLCAX1,HH1,XN,AX1,XLAG,UNIT,TIMLAG,
        CBASCON1,VSL1,SLP
7000 FORMAT(10X,'WATERCOURSE LENGTH = ',F11.3,' M ',/,
        C10X,'LENGTH FROM CONCENTRATION POINT TO CENTROID =',
        CF11.3,' M ',/,10X,'ELEVATION VARIATION ALONG ',

```

```

C'WATERCOURSE = ',F11.3,' M',/,10X,
C'BASIN FACTOR = ',F5.3,/,
C10X,'WATERSHED AREA = ',F11.3,'SQUARE KM',
C/,10X,'WATERCOURSE "LAG" TIME = ',F8.3,' HOURS',/,
C10X,'UNIT HYDROGRAPH TIME UNIT = ',F7.3,' MINUTES',/,
C10X,'UNIT INTERVAL PERCENTAGE OF LAG-TIME = ',F7.3,/,
C10X,'BASE FLOW = ',F7.3,' CMS/SQUARE-KM',/,10X,
C 'MAXIMUM WATERSHED LOSS RATE(CM/HOUR) = ',F6.3,/,
C10X,'LOW LOSS RATE PERCENTAGE(DECIMAL) = ',F5.3)
GO TO 7002
7001 WRITE(13,5001)XLX,XLCAX,HH,XN,AX,XLAG,UNIT,TIMLAG,
CBASCON,VSL,SLP
5001 FORMAT(10X,'WATERCOURSE LENGTH = ',F11.3,' FEET ',/,
C10X,'LENGTH FROM CONCENTRATION POINT TO CENTROID = ',
CF11.3,' FEET',/,10X,'ELEVATION VARIATION ALONG ',
C'WATERCOURSE = ',F11.3,'FEET',/,10X,
C'BASIN FACTOR = ',F5.3,/,
C10X,'WATERSHED AREA = ',F11.3,'ACRES'
C/,10X,'WATERCOURSE "LAG" TIME = ',F8.3,' HOURS',/,
C10X,'UNIT HYDROGRAPH TIME UNIT = ',F7.3,' MINUTES',/,
C10X,'UNIT INTERVAL PERCENTAGE OF LAG-TIME = ',F7.3,/,
C10X,'BASE FLOW = ',F7.3,' CFS/SQUARE-MILE',/,10X,
C 'MAXIMUM WATERSHED LOSS RATE(INCH/HOUR) = ',F6.3,/,
C10X,'LOW LOSS RATE PERCENTAGE(DECIMAL) = ',F5.3)
7002 IF(KODE1.EQ.1)WRITE(13,8200)
IF(KODE1.EQ.2)WRITE(13,8202)
IF(KODE1.EQ.3)WRITE(13,8204)
IF(KODE1.EQ.4)WRITE(13,8205)
IF(KODE1.EQ.5)WRITE(13,82041)PV,PF,PM,PD
IF(KODE1.EQ.6)WRITE(13,82042)
82041 FORMAT(10X,'VALLEY "S"-CURVE PERCENTAGE(DECIMAL NOTATION) = ',
CF6.3,/,10X,'FOOTHILL "S"-CURVE PERCENTAGE(DECIMAL NOTATION) = ',
C,F6.3,/,10X,'MOUNTAIN "S"-CURVE PERCENTAGE(DECIMAL NOTATION) = ',
C,F6.3,/,10X,'DESERT "S"-CURVE PERCENTAGE(DECIMAL NOTATION) = ',
CF6.3,/)
8205 FORMAT(10X,'DESERT S-GRAPH SELECTED',/)
8200 FORMAT(10X,'VALLEY S-GRAPH SELECTED',/)
8202 FORMAT(10X,'FOOTHILL S-GRAPH SELECTED',/)
8204 FORMAT(10X,'MOUNTAIN S-GRAPH SELECTED',/)
82042 FORMAT(10X,'U.S. SOIL CONSERVATION SERVICE S-GRAPH SELECTED',/)
WRITE(*,8209)R5,R30
IF(KCODE2.NE.1)GO TO 7003
R51=R5*2.54
R301=R30*2.54
R11=R1*2.54
R31=R3*2.54
R61=R6*2.54
R241=R24*2.54
WRITE(13,7004)R51,R301
7004 FORMAT(10X,'SPECIFIED PEAK 5-MINUTE RAINFALL(CM) = ',F5.2,/,
C10X,'SPECIFIED PEAK 30-MINUTES RAINFALL(CM) = ',F5.2)
WRITE(*,7005)R11,R31,R61,R241,KTYPE
WRITE(13,7005)R11,R31,R61,R241,KTYPE
7005 FORMAT(10X,'SPECIFIED PEAK 1-HOUR RAINFALL(CM) = ',F5.2,/,
C10X,'SPECIFIED PEAK 3-HOUR RAINFALL(CM) = ',F5.2,/,

```

```

C10X,'SPECIFIED PEAK 6-HOUR RAINFALL(CM) = ',F5.2,/,
C10X,'SPECIFIED PEAK 24-HOUR RAINFALL(CM) = ',F5.2,/,
C10X,'>>>HYDROGRAPH MODEL<<< #',I1,' SPECIFIED*')
GO TO 7006
7003 WRITE(13,8209)R5,R30
8209 FORMAT(10X,'SPECIFIED PEAK 5-MINUTE RAINFALL(INCH) = ',F5.2,/,
C10X,'SPECIFIED PEAK 30-MINUTES RAINFALL(INCH) = ',F5.2)
WRITE(*,8206)R1,R3,R6,R24,KTYPE
WRITE(13,8206)R1,R3,R6,R24,KTYPE
8206 FORMAT(10X,'SPECIFIED PEAK 1-HOUR RAINFALL(INCH) = ',F5.2,/,
C10X,'SPECIFIED PEAK 3-HOUR RAINFALL(INCH) = ',F5.2,/,
C10X,'SPECIFIED PEAK 6-HOUR RAINFALL(INCH) = ',F5.2,/,
C10X,'SPECIFIED PEAK 24-HOUR RAINFALL(INCH) = ',F5.2,/,
C10X,'>>>HYDROGRAPH MODEL<<< #',I1,' SPECIFIED*')
7006 IF(DAOPT.EQ.1.)WRITE(13,6665)
IF(DAOPT.EQ.2.)WRITE(13,6666)
WRITE(*,6664)FX5,FX30,FX1,FX3,FX6,FX24
WRITE(13,6664)FX5,FX30,FX1,FX3,FX6,FX24

C
C USER TIME LIMITS
C IF(TIME2.EQ.31.)GO TO 957
WRITE(*,958)TIME1,TIME2
WRITE(13,958)TIME1,TIME2
958 FORMAT(/10X,'RUNOFF HYDROGRAPH LISTING LIMITS:',
1/10X,'MODEL TIME(HOURS) FOR BEGINNING OF RESULTS',
2'=',F8.2,
3/10X,'MODEL TIME(HOURS) FOR END OF RESULTS ',
4'=',F8.2)

C
XR=VSL*UNIT/60.

C
C DETERMINATION OF UNIT HYDROGRAPH ORDINATES
C
IF(KODE1.LT.5 .OR. KODE1.EQ.6)GO TO 94

C
C=====LINEAR WEIGHTING OF S-CURVES
NLAG=0
DO 98 KLAG=1,4
PLAG=PV
IF(KLAG.EQ.2)PLAG=PF
IF(KLAG.EQ.3)PLAG=PM
IF(KLAG.EQ.4)PLAG=PD
IF(KLAG.EQ.0)GO TO 98
CALL SUBSB(TIMLAG,PERCNT,KLAG,NUMBER)
IF(NUMBER.GT.NLAG)NLAG=NUMBER
DO 99 I=1,150
IF(PERCNT(I).EQ.0)PERCNT(I)=100.
H(I)=H(I)+PERCNT(I)*PLAG
99 PERCNT(I)=0.
98 CONTINUE

C
C UNIT HYDROGRAPH DETERMINATION
C
NUMBER=NLAG
C

```

```

      DO 97 I=1,150
      IF(I.LE.NLAG)PERCNT(I)=H(I)
97  H(I)=0
      GO TO 96
C
      94 CALL SUBSB(TIMLAG,PERCNT,KODE1,NUMBER)
C
      96 SUM=0.
      IF(NUMBER.GE.150)WRITE(13,8207)
8207  FORMAT(3X,'UNIT HYDROGRAPH TERMINATED AFTER 150 UNIT INTERVALS',/)
      IF(NUMBER.GT.150)NUMBER=150
C
      DO 8208 I=1,NUMBER
      UH(I)=(PERCNT(I)-SUM)*XK
      SUM=PERCNT(I)
      IF(UH(I).LT.0.)UH(I)=0.
8208  CONTINUE
C
      WRITE(*,5002)
      WRITE(13,5002)
5002  FORMAT(23X,'UNIT HYDROGRAPH DETERMINATION',/)
      WRITE(*,5003)
      WRITE(13,5003)
5003  FORMAT(76('-',))
      IF(KCODE2.NE.1)GO TO 7007
      WRITE(*,5004)
      WRITE(13,7008)
7008  FORMAT(5X,'INTERVAL',10X,'"S" GRAPH',10X,'UNIT HYDROGRAPH',/,
      6X,'NUMBER',10X,'MEAN VALUES',10X,'ORDINATES(CMS)',)
      WRITE(*,5003)
      WRITE(13,5003)
      WRITE(*,5005)(I,PERCNT(I),UH(I)*0.028338,I=1,NUMBER)
      WRITE(13,5005)(I,PERCNT(I),UH(I)*0.028338,I=1,NUMBER)
      GO TO 7020
7007  WRITE(13,5004)
5004  FORMAT(5X,'INTERVAL',10X,'"S" GRAPH',10X,'UNIT HYDROGRAPH',/,
      6X,'NUMBER',10X,'MEAN VALUES',10X,'ORDINATES(CFS)',)
      WRITE(*,5003)
      WRITE(13,5003)
      WRITE(*,5005)(I,PERCNT(I),UH(I),I=1,NUMBER)
      WRITE(13,5005)(I,PERCNT(I),UH(I),I=1,NUMBER)
5005  FORMAT(7X,I3,15X,F7.3,13X,F10.3)
C
7020  WRITE(*,5000)
      WRITE(13,5000)
5050  FORMAT(A1)
      IF(KSOIL.EQ.1)WRITE(13,5000)
      IF(KCODE2.NE.1)GO TO 7009
      IF(KSOIL.EQ.1)WRITE(13,7010)
7010  FORMAT(10X,'UNIT',14X,'UNIT',12X,'UNIT',14X,'EFFECTIVE',/,
      9X,'PERIOD',11X,'RAINFALL',7X,'SOIL-LOSS',12X,'RAINFALL',/,
      8X,'(NUMBER)',10X,'( CM )',8X,'( CM )',12X,'( CM )')
      IF(KSOIL.EQ.1)WRITE(13,5003)
      GO TO 7011
7009  IF(KSOIL.EQ.1)WRITE(13,5020)

```

```

      IF(KSOIL.EQ.1)WRITE(13,5003)
5020  FORMAT(10X,'UNIT',14X,'UNIT',12X,'UNIT',14X,'EFFECTIVE',/,
      C9X,'PERIOD',11X,'RAINFALL',7X,'SOIL-LOSS',12X,'RAINFALL',/,
      C8X,'(NUMBER)',10X,'(INCHES)',8X,'(INCHES)',12X,'(INCHES)')

```

C

C

C 24-HOUR STORM RAINFALL PATTERN

C

```

7011  R5A=R5*FX5
      R30A=R30*FX30
      R1A=R1*FX1
      R3A=R3*FX3
      R6A=R6*FX6
      R24A=R24*FX24

```

C

C

```

      A=(ALOG(R30A)-ALOG(R5A))/(ALOG(.5)-ALOG(.0833))
      B=ALOG(R5A)-A*ALOG(.0833)
      R(193)=RR(.0833)
      R(194)=RR(.1667)-RR(.0833)
      R(195)=RR(.25)-RR(.1667)
      R(192)=RR(.3333)-RR(.25)
      R(196)=RR(.41667)-RR(.3333)
      R(191)=RR(.5)-RR(.41667)
      A=(ALOG(R1A)-ALOG(R30A))/(ALOG(1.)-ALOG(.5))
      B=ALOG(R30A)-A*ALOG(.5)
      R(197)=RR(.5833)-RR(.5)
      R(190)=RR(.6667)-RR(.5833)
      R(198)=RR(.75)-RR(.6667)
      R(189)=RR(.8333)-RR(.75)
      R(188)=RR(.9167)-RR(.8333)
      R(187)=RR(1.)-RR(.9167)

```

C.....REMAINING PART OF PEAK 3-HOUR STORM

```

      A=(ALOG(R3A)-ALOG(R1A))/(ALOG(3.)-ALOG(1.))
      B=ALOG(R1A)-A*ALOG(1.)
      RRSAVE=R1A

```

C

```

      DO 1001 J=1,12
      XJ=J
      DT=XJ*.1667
      T=1.+DT
      RRNEW=RR(T)
      DR=(RRNEW-RRSAVE)/2.
      R(198+J)=DR
      R(187-J)=DR

```

1001 RRSAVE=RRNEW

C

C REMAINING PART OF PEAK 6-HOUR STORM

C

```

      A=(ALOG(R6A)-ALOG(R3A))/(ALOG(6.)-ALOG(3.))
      B=ALOG(R3A)-A*ALOG(3.)
      RRSAVE=R3A

```

C

```

      DO 1010 J=1,18
      XJ=J

```

```

DT=XJ*.1667
T=3.+DT
RRNEW=RR(T)
DR=(RRNEW-RRSAVE)/2.
R(210+J)=DR
R(175-J)=DR
1010 RRSave=RRNEW
C
C   REMAINING PART OF PEAK 24-HOUR STORM
A=(ALOG(R24A)-ALOG(R6A))/(ALOG(24.)-ALOG(6.))
B=ALOG(R6A)-A*ALOG(6.)
RRSAVE=R6A
C
DO 1020 J=1,60
XJ=J
DT=XJ*.1667
T=6.+DT
RRNEW=RR(T)
DR=(RRNEW-RRSAVE)/2.
R(228+J)=DR
R(157-J)=DR
1020 RRSave=RRNEW
C
DO 1030 J=1,96
XJ=J
DT=XJ*.08333
T=16.+DT
RRNEW=RR(T)
DR=RRNEW-RRSAVE
R(97-J)=DR
1030 RRSave=RRNEW
C
C
C   ADJUST R-ARRAY FOR LARGER UNIT INTERVALS
C
C   NI=NUMBER OF STORM HYDROGRAPH INTERVALS
NI=288
IF(KTYPE.EQ.1)GO TO 1050
K=KTYPE
IF(KTYPE.EQ.5)K=6
IF(KTYPE.EQ.6)K=12
NI=288/K
DO 1040 I=1,NI
TEMP=0.
II=(I-1)*K
DO 1035 J=1,K
1035 TEMP=TEMP+R(II+J)
1040 R(I)=TEMP
1050 CONTINUE
C
C   ADJUST FOR CONSTANT SOIL LOSS
C
XTOTAL=0.
TEMPS=UNIT/60.

```

```

XRA=XR
DO 400 I=1,NI
XLOSS=R(I)*SLP
IF(XLOSS.GT.XRA) XLOSS=XRA
TEMP=R(I)
R(I)=R(I)-XLOSS
XTOTAL=XTOTAL+XLOSS
IF(KCODE2.NE.1)GO TO 7012
TEMP1=TEMP*2.54
RAIN(I)=R(I)*2.54
XLOSS1=XLOSS*2.54
C
  IF(KSOIL.EQ.1)WRITE(13,5021)I,TEMP1,XLOSS1,RAIN(I)
  GO TO 400
7012 IF(KSOIL.EQ.1)WRITE(13,5021)I,TEMP,XLOSS,R(I)
5021 FORMAT(10X,13,13X,F6.4,11X,F6.4,14X,F6.4)
400 CONTINUE
C
C DETERMINE STORM RUNOFF HYDROGRAPH
C
  INTERV=NUMBER+NI-1
  IF(INTERV.GT.440)WRITE(13,432)
  IF(INTERV.GT.439)INTERV=439
432 FORMAT(3X,'RUNOFF HYDROGRAPH TERMINATED AFTER 440 UNIT',/,
C3X,'INTERVALS. SUGGEST USING A LARGER UNIT INTERVAL.')
  DO 600 I=1,INTERV
  M=I
  N=1+M
  DO 500 J=1,M
  K=N-J
  IF(J.GT.NI)GO TO 500
  IF(K.GT.NUMBER)GO TO 500
  H(M)=H(M)+R(J)*UH(K)
500 CONTINUE
  H(M)=H(M)+BASFLO
600 CONTINUE
C
C COMPUTE SUM HYDROGRAPH
C
  SUM=0.0
  XMAX=0.
  DO 700 I=1,INTERV
  IF(H(I).LT.0.)H(I)=0.
  IF(H(I).GT.XMAX)XMAX=H(I)
  SUM=SUM+H(I)
700 CONTINUE
  SUM=SUM*UNIT*60./43560.
  XTOTAL=XTOTAL/12.*AREA
  IF(KCODE2.NE.1)GO TO 7013
  XTOTA1=XTOTAL/836.4
  SUM1=SUM/836.4
  WRITE(13,7014)XTOTA1,SUM1
7014 FORMAT(5X,'TOTAL SOIL-LOSS VOLUME(SQUARE-KM-M) =',F12.4,/,
C5X,'TOTAL STORM RUNOFF VOLUME(SQUARE-KM-M) =',F12.4)
  GO TO 7021

```

```

C
7013 WRITE(*,5003)
      WRITE(13,5003)
      WRITE(*,6008)XTOTAL,SUM
      WRITE(13,6008)XTOTAL,SUM
6008 FORMAT(5X,'TOTAL SOIL-LOSS VOLUME(ACRE-FEET) =',F12.4,/,
      C5X,'TOTAL STORM RUNOFF VOLUME(ACRE-FEET) =',F12.4)
C
7021 IF(XMAX.LT.100.)GO TO 8000
      I=XMAX/100.
      II=I+1
      XMAX=II
      XMAX=XMAX*100.
      GO TO 8100
8000 I=XMAX/10.
      II=10*(I+1)
      XMAX=II
8100 CONTINUE
      WRITE(*,5003)
      WRITE(13,5003)
C
      CALL OASB(KTYPE,KCODE2,H,INTERV,XMAX,UNIT,SUM,TIME1,TIME2)
C
10000 CONTINUE
C
      RETURN
      END
$DEBUG
      SUBROUTINE SUBSB(TIMLAG,PERCNT,KODE1,NUMBER)
C
C
C      THIS SUBROUTINE DETERMINES THE PERCNTAGES OF DICHARGE
C      FACTOR FOR THE VARIOUS ZONE CLASSIFICATIONS.
C
C
C      REAL MNT
C      DIMENSION PERCNT(150),VAL(31,2),FOOT(31,2),MNT(33,2),SCS(33,2)
C      DIMENSION DESERT(33,2)
C
      DATA VAL/0.,15.0,25.,35.0,50.,65.0,75.,100.,115.,125.,
C140.0,150.0,165.0,175.0,200.0,225.0,250.,275.,300.,325.,350.,
C375.,400.,450.,500.,550.,600.,650.,700.,750.,99999.,0.0,2.6,
C5.0,8.6,15.5,25.0,32.0,50.0,57.9,62.0,66.8,69.5,72.6,74.3,
C78.0,81.0,83.5,85.7,87.5,89.0,90.5,91.6,92.7,94.3,95.8,
C96.9,97.8,98.5,99.0,99.5,100. /
      DATA FOOT/0.,15.0,25.,35.0,50.,60.0,75.,85.0,90.0,95.0,
C100.0,110.0,125.0,140.0,150.0,175.0,200.0,225.0,250.0,
C275.0,300.0,325.0,350.0,375.0,400.0,450.0,500.0,550.0,
C600.0,650.0,99999.,0.0,1.9,3.8,6.0,10.3,14.0,21.7,29.0,
C34.2,43.3,50.0,56.9,63.8,69.0,71.9,77.8,82.4,86.0,89.0,
C91.4,93.4,95.0,96.2,97.2,97.9,98.5,99.0,99.3,99.7,
C99.9,100. /
      DATA MNT/0.,15.0,25.,35.0,40.0,50.,65.0,75.0,90.0,
C100.0,115.0,125.0,140.0,150.0,175.0,200.0,225.0,250.0,
C275.0,300.0,325.0,350.0,375.0,400.0,450.0,500.0,

```

C550.0,600.0,650.0,700.0,750.0,800.0,99999.0,0.0,
 C3.3,6.7,10.6,13.4,21.0,33.0,39.3,46.3,50.0,54.2,
 C56.7,59.8,61.8,65.8,69.2,72.2,74.8,77.0,79.0,80.7,
 C82.2,83.5,84.8,86.9,88.9,90.5,92.0,93.3,94.5,95.5,
 C96.4,100. /
 DATA DESERT/0.,12.5,25.0,37.5,50.0,62.5,75.0,87.5,100.,
 C112.5,125.,137.5,150.,162.5,175.,187.5,200.,225.,250.,275.,
 C300.,325.,350.,375.,400.,450.,500.,550.,600.,700.,800.,1000.,
 C9999.,0.,1.1,3.2,6.3,10.5,18.5,31.3,42.0,50.0,56.5,61.3,65.2,
 C68.5,71.5,74.0,76.2,78.3,81.6,84.3,86.7,88.7,90.2,91.6,92.8,
 C93.9,95.6,96.9,97.8,98.3,99.5,99.9,99.99,100./
 DATA SCS/0.,8.6,17.2,25.9,34.5,43.1,51.7,60.3,69.,77.6,
 C86.2,94.8,103.4,112.1,120.7,129.3,137.9,146.5,155.2,
 C163.8,172.4,189.6,206.9,224.1,241.4,258.6,275.8,293.1,
 C310.3,327.6,344.8,387.9,999.,0.,.1,.6,1.2,3.5,6.5,10.7,
 C16.3,22.8,30.,37.5,45.,52.2,58.9,65.,70.,75.1,79.,82.2,
 C84.9,87.1,90.8,93.4,95.3,96.7,97.7,98.4,98.9,99.3,99.5,99.7,
 C99.9,100./

C
 C KODE1=1 : VALLEY
 C KODE1=2 : FOOTHILL
 C KODE1=3 : MOUNTAIN
 C KODE1=4 : DESERT
 C KODE1=5 : NOT USED DIRECTLY. LINEAR INTERPOLATION OF 1-4
 C KODE1=6 : SCS
 C TIMELAG=LAG
 C TIME=INCREMENT OF LAG
 C NUMBER=INDEX OF PERCENT VECTOR
 C

ANEW=0.

AOLD=0.

K=1

NUMBER=1

TIME=TIMLAG

10 K=K+1

IF(NUMBER.GE.151)GO TO 1000

N=K-1

C

GO TO(100,200,300,350,355,355),KODE1

C

100 CONTINUE

C

C

INTEGRATE "S" GRAPH IN ORDER TO DETERMINE UH(I)

C

TEMP=0.5*(VAL(K,2)+VAL(N,2))*

C(VAL(K,1)-VAL(N,1))

ANEW=ANEW+TEMP

IF(TIME.GT.VAL(K,1))GO TO 10

Y=VAL(K,2)

X=VAL(N,2)

DEL=(TIME-VAL(N,1))/(VAL(K,1)-VAL(N,1))

B=VAL(K,1)-VAL(N,1)

GO TO 400

200

CONTINUE

TEMP=0.5*(FOOT(K,2)+FOOT(N,2))*

```

C(FOOT(K,1)-FOOT(N,1))
ANEW=ANEW+TEMP
IF(TIME.GT.FOOT(K,1))GO TO 10
C
Y=FOOT(K,2)
X=FOOT(N,2)
DEL=(TIME-FOOT(N,1))/(FOOT(K,1)-FOOT(N,1))
B=FOOT(K,1)-FOOT(N,1)
GO TO 400
300 CONTINUE
TEMP=0.5*(MNT(K,2)+MNT(N,2))*
C(MNT(K,1)-MNT(N,1))
ANEW=ANEW+TEMP
IF(TIME.GT.MNT(K,1))GO TO 10
Y=MNT(K,2)
X=MNT(N,2)
DEL=(TIME-MNT(N,1))/(MNT(K,1)-MNT(N,1))
B=MNT(K,1)-MNT(N,1)
C
GO TO 400
350 CONTINUE
TEMP=0.5*(DESERT(K,2)+DESERT(N,2))*
C(DESERT(K,1)-DESERT(N,1))
ANEW=ANEW+TEMP
IF(TIME.GT.DESERT(K,1))GO TO 10
Y=DESERT(K,2)
X=DESERT(N,2)
DEL=(TIME-DESERT(N,1))/(DESERT(K,1)-DESERT(N,1))
B=DESERT(K,1)-DESERT(N,1)
GO TO 400
C
355 CONTINUE
C
C SCS METHOD
C
TEMP=0.5*(SCS(K,2)+SCS(N,2))*(SCS(K,1)-SCS(N,1))
ANEW=ANEW+TEMP
IF(TIME.GT.SCS(K,1))GO TO 10
Y=SCS(K,2)
X=SCS(N,2)
DEL=(TIME-SCS(N,1))/(SCS(K,1)-SCS(N,1))
B=SCS(K,1)-SCS(N,1)
C
400 CONTINUE
DEL=DEL*(Y-X)
XX=X+DEL
C
C ADJUST INTEGRATION FOR INTERPOLATION
C
DELA=0.5*(Y+XX)*(1.-DEL/(Y-X))*B
ANEW=ANEW-DELA
PERCNT(NUMBER)=(ANEW-AOLD)/TIMLAG
NUMBER=NUMBER+1
AOLD=ANEW
ANEW=ANEW-.5*(X+XX)*DEL/(Y-X)*B

```

```

TIME=TIME+TIMLAG
K=K-1
IF(NUMBER.EQ.2)GO TO 10
DELX=PERCNT(NUMBER-1)-PERCNT(NUMBER-2)
IF(DELX.LE..51)GO TO 1000
GO TO 10
1000 CONTINUE
NUMBER=NUMBER-1
IF(NUMBER.GE.150)WRITE(7,1001)NUMBER
1001 FORMAT(7X,'UNIT-HYDROGRAPH TERMINATEED AFTER',I4,'INTERVALS')
C
C   FINISH-OFF HYDROGRAPH
C
IF(NUMBER.GE.150)GO TO 1250
NNUM=150-NUMBER
XNUM=NNUM
REM=100.-PERCNT(NUMBER)
REM1=REM/XNUM
DELX=.5
IF(REM1.LT.DELX)GO TO 1150
DO 1140 K=1,NNUM
1140 PERCNT(NUMBER+K)=PERCNT(NUMBER+K-1)+REM1
GO TO 1200
1150 XNUM=REM/DELX+1.
NNUM=XNUM
DO 1160 K=1,NNUM
1160 PERCNT(NUMBER+K-1)=PERCNT(NUMBER+K-2)+DELX
NNUM=NNUM-1
1200 NUMBER=NUMBER+NNUM
IF(PERCNT(NUMBER).GE.100.)PERCNT(NUMBER)=100.
IF(NUMBER.GE.100)GO TO 1250
NUMBER=NUMBER+1
PERCNT(NUMBER)=100.
1250 CONTINUE
C
RETURN
END
$DEBUG
SUBROUTINE OASB(KTYPE,KCODE2,H,INTERV,XMAX,UNIT,SUM,TIME1,TIME2)
C
DIMENSION H(440),H1(440)
INTEGER LETM,BLANK,DOT,CROSS,DASH,LINE(41)
DATA LETM,BLANK,DOT,CROSS,DASH,LINE/'V',' ','.',',','Q','I',41*' '/
C
C
5050 FORMAT(1X,A1)
WRITE(*,101)
WRITE(13,101)
WRITE(*,10)
WRITE(13,10)
10 FORMAT(/,25X,'2 4 - H O U R S T O R M',/,
21X,'R U N O F F H Y D R O G R A P H',/)
100 CONTINUE
WRITE(*,101)
WRITE(13,101)

```

```

101 FORMAT(76('*'))
102 FORMAT(76('-'))
STEP=5.*60./43560.
WRITE(*,103)
IF(KCODE2.NE.1)GO TO 7026
WRITE(13,7024)
7024 FORMAT(18X,'HYDROGRAPH IN FIVE-MINUTE INTERVALS(CMS)',/)
GO TO 7025
7026 WRITE(13,103)
103 FORMAT(18X,'HYDROGRAPH IN FIVE-MINUTE INTERVALS(CFS)',/)
7025 WRITE(*,102)
WRITE(13,102)
F=40./XMAX
FMASS=40./SUM
T0=0.0
T1=XMAX/4.
T2=2.*T1
T3=3.*T1
IF(KCODE2.NE.1)GO TO 7015
T1=T1*0.0283386
T2=2.*T1
T3=3.*T1
XMAX=XMAX*0.0283386
WRITE(*,7016)T0,T1,T2,T3,XMAX
WRITE(13,7016)T0,T1,T2,T3,XMAX
7016 FORMAT(2X,'TIME(HRS) VOLUME(SQR.KM-M) Q(CMS) ',F2.0,1X,4F10.1)
GO TO 7017
7015 WRITE(13,105)T0,T1,T2,T3,XMAX
105 FORMAT(2X,'TIME(HRS) VOLUME(AF) Q(CFS) ',F2.0,1X,4F10.1)
7017 WRITE(*,102)
WRITE(13,102)
XMASS=0.

C
GO TO(141,142,143,146,147,148),KTYPE
141 KNUM=1
GO TO 144
142 KNUM=2
GO TO 144
143 KNUM=3
GO TO 144
146 KNUM=4
GO TO 144
147 KNUM=6
GO TO 144
148 KNUM=12
144 TIME=0.

C
C OUTPUT GRAPH LOOP
C
DO 200 I=1,INTERV
TEST=H(I)*F
XMASS=XMASS+H(I)*STEP
LINE(1)=DOT
LINE(11)=DOT
LINE(21)=DOT

```

```

LINE(31)=DOT
LINE(41)=DOT
J=TEST+1
JMASS=XMASS*FMASS+1
LINE(JMASS)=LETM
LINE(J)=CROSS
IF(KNUM.EQ.1)GO TO 400

```

C

```

320 DO 350 K=1,KNUM
    TIME=TIME+.083333
    IF(K.EQ.1)GO TO 7023
    XMASS=XMASS+H(I)*STEP
    JMASS=XMASS*FMASS+1
    LINE(JMASS)=LETM
    LINE(J)=CROSS
7023 IF(KCODE2.NE.1)GO TO 349
    XMAS1=XMASS/836.4
    H1(I)=H(I)*0.0283386
    IF(TIME.GE.TIME1.AND.TIME.LE.TIME2)WRITE(13,210)TIME,XMAS1,
    1H(I),LINE
    GO TO 350
349 IF(TIME.GE.TIME1.AND.TIME.LE.TIME2)WRITE(13,210)TIME,XMASS,
    1H(I),LINE
350 LINE(JMASS)=BLANK
    GO TO 215

```

C

```

400 CONTINUE
    TIME=TIME+.083333
    IF(KCODE2.NE.1)GO TO 7022
    XMAS1=XMASS/836.4
    H1(I)=H(I)*0.0283386
    IF(TIME.GE.TIME1.AND.TIME.LE.TIME2)WRITE(13,210)TIME,XMAS1,
    1H(I),LINE
    GO TO 215
7022 IF(TIME.GE.TIME1.AND.TIME.LE.TIME2)WRITE(13,210)TIME,XMASS,
    1H(I),LINE
210 FORMAT(2X,F7.3,F12.4,1X,F9.2,2X,41A1)
215 LINE(J)=BLANK
    LINE(JMASS)=BLANK
200 CONTINUE
    WRITE(*,101)
    WRITE(13,101)

```

C

```

    WRITE(8,201)INTERV
201 FORMAT(I12)
    DO 203 I=1,INTERV
    WRITE(8,202)H1(I)
202 FORMAT(F12.2)
203 CONTINUE
    CLOSE(3)
    CLOSE(8)
    CLOSE(13)
    RETURN
    END

```

\$DEBUG

```

SUBROUTINE ROUT
C
C   RESERVOIR ROUTING
C   BY FINITE DIFFERENCE
C
  DIMENSION STAGE(500),FLIN(500)
  DIMENSION STOR(20),ELV(20),A22(5),HWL1(20),WL1(20),A44(5)
  COMMON/ROUT1/STOR,ELV,A22
  COMMON/ROUT3/HWL1,WL1,A44
  REAL LEVEL

C
C   DATA INPUT
C   NO           =number of the interval data from reservoir capacity curve
C   NO1          =number increment of spillway length
C   DW           =increment of spillway length.....m
C   ELV          =elevation at any reservoir capacity .....m
C   STOR         =reservoir capacity.....m^3
C   WL          =spillway length.....m
C   CREST        =level of dam crest.....m
C   HWL          =allowable of high water level.....m
C   SPILL        =level of spillway crest.....m
C   LEVEL        =water level at initial time.....m
C   DT           =time interval.....hr
C   DIA          =diameter of outlet work.....m
C   NCON         =number of outlet work
C   OUT          =level of outlet work.....m
C   NO2          =number of the interval data from inflow hydrograph
C   FLIN         =inflow hydrograph.....cms
C
  OPEN(4,FILE='ROUT.PRN')
  OPEN(7,FILE='STOR.DAT')
  OPEN(8,FILE='ROUT.DAT')
  OPEN(9,FILE='CHUTE.DAT')
  OPEN(14,FILE='ROUT.RES')
  READ(4,30)CREST,HWL,SPILL,LEVEL,OUT
  WRITE(*,30)CREST,HWL,SPILL,LEVEL,OUT
30  FORMAT(//////////5F12.2)
  READ(4,40)NO1,WL,DW
  WRITE(*,40)NO1,WL,DW
40  FORMAT(I12,2F12.2)
  READ(4,50)NCON,DIA,KENDT
  WRITE(*,50)NCON,DIA,KENDT
50  FORMAT(I12,F12.2,I12)
  READ(7,10)NO
  WRITE(*,10)NO
10  FORMAT(I12)
  READ(7,20)(STOR(I),ELV(I),I=1,NO)
  WRITE(*,20)(STOR(I),ELV(I),I=1,NO)
20  FORMAT(2F12.2)
11  CONTINUE
  READ(8,21)NO2
  WRITE(*,21)NO2
21  FORMAT(I12)
  READ(8,22)(FLIN(I),I=1,NO2)
  WRITE(*,22)(FLIN(I),I=1,NO2)

```

```

22 FORMAT(F12.2)
C
C   CALL SUBROUTINE LST TO FITT STORAGE CAPACITY CURVE
C
      N1=NO-1
      M1=2
      CALL LST(ELV,STOR,N1,M1,A22)
C
C   COMPUTE OUTFLOW HYDROGRAPH INTERVAL 1.  HR
C
      DT=1.
      DT=DT*3600.
      PI=22/7.
      ELV1=LEVEL
120 IF(OP.NE.1.)GO TO 121
      NO1=1
121 DO 620 K=1,NO1
      LEVEL=ELV1
      WRITE(14,55)K
      WRITE(*,55)K
55  FORMAT(10X,'COMPUTATION FOR OPENNING SPILLWAY NO =',I3)
      TIME=0.
      HEAD1=LEVEL-OUT
      IF(HEAD1.GT.0.)GO TO 102
      HEAD1=0.
C
C   DISCHARGE DRAINAGE FROM OUTLET WORK
C   AT INITIAL TIME
C
102 QOUT=((DIA**2)*PI*0.6*((HEAD1*2*9.807)**0.5)/4)*NCON
      HEAD2=LEVEL-SPILL
      IF(HEAD2.GT.0.)GO TO 101
      HEAD2=0.
C
C   DISCHARGE SPILL FROM SPILLWAY
C   AT INITIAL TIME
C
101 QSPIL=1.73*WL*(HEAD2**1.5)
      OUT1=QSPIL+QOUT
      STOR1=(A22(1)+A22(2)*LEVEL+A22(3)*(LEVEL**2))*1000.
C
C   SOLVED EQUATION
C   AT TIME +DELTA TIME BY MODIFIED NEWTON RALPHSON
C   WATER LEVEL AT INITIAL TIME =LEVEL
C
      DO 63 JJ=1,KENDT
      FLOW1=FLIN(JJ)
      FLOW2=FLIN(JJ+1)
      FT=0.01
      ERR=0.001
C
C   TRIAL LEVEL AT THE FIRST TIME
C
      LEVEL=LEVEL+0.15
6  G=LEVEL

```

```

WRITE(*,61)LEVEL
61 FORMAT(2X,'&& TRIAL WATER LEVEL ??=',F6.2)
XT=2.
71 HEAD1=LEVEL-OUT
PI=22/7.
IF(HEAD1.GT.0.)GO TO 7
HEAD1=0.
7 OCON=((2*9.807*HEAD1)**0.5)*PI*(DIA**2)*NCON/4
HEAD2=LEVEL-SPILL
IF(HEAD2.GT.0.)GO TO 8
HEAD2=0.
8 OSPILL=1.73*WL*(HEAD2**1.5)
OUT2=OCON+OSPILL
STOR2=(A22(1)+A22(2)*LEVEL+A22(3)*(LEVEL**2))*1000.
HI=((FLOW1+FLOW2)/2-(OUT1+OUT2)/2)*DT-STOR2+STOR1
LEVEL=LEVEL+FT
XT=XT-1
IF(XT.EQ.0.)GO TO 81
HT1=HI
GO TO 71
81 HI=(HI-HT1)/FT
HT1=G-HT1/HI
IF(ABS(HT1-G).LT.ERR) GO TO 91
LEVEL=HT1
GO TO 6
91 LEVEL=G
STAGE(JJ)=G
OUT1=OUT2
STOR1=STOR2
WRITE(14,57)TIME,STAGE(JJ)
WRITE(*,57)TIME,STAGE(JJ)
57 FORMAT(10X,'TIME HR.',F6.2,10X,'WATER LEVEL =',F6.2)
TIME=TIME+DT/3600.
63 CONTINUE
C WRITE(*,58)
C 58 FORMAT(10X,'****DRAINAGE TIME IS OVER 500 HRS.****')
IF(OP.EQ.1.)GO TO 800
XMAX=STAGE(1)
DO 600 I=2,KENDT
IF(XMAX-STAGE(I))610,600,600
610 XMAX=STAGE(I)
600 CONTINUE
WL1(K)=WL
HWL1(K)=XMAX
WL=WL+DW
620 CONTINUE
WRITE(14,64)(WL1(II),HWL1(II),II=1,NO1)
64 FORMAT(1X,'$$ TEST OPENNING SPILLWAY =',F8.2,5X,F8.2)
C
C INTERPOLATE OPTIMAL OPENNING SPILLWAY AT HIGH WATER LEVEL
C
IF((HWL.GT.HWL1(1)).AND.(HWL.LT.HWL1(NO1)))GO TO 66
DO 65 II=2,NO1
IF(HWL.LT.HWL1(II)) GO TO 65
WL=(HWL-HWL1(II-1))*(WL1(II)-WL1(II-1))/(HWL1(II)-HWL1(II-1))

```

```

      $+WL1(II-1)
      GO TO 68
65  CONTINUE
66  WRITE(*,67)
67  FORMAT(1X,'DATA OPENNING SPILLWAY TO INTERPOLATE NOT ENOUGH')
      STOP
68  WRITE(14,801)WL
      WRITE(*,801)WL
801  FORMAT(10X,'SPILLWAY OPENNING ( M ) =',F6.2)
      OP=1.
      GO TO 120
800  WRITE(14,810)
      WRITE(*,810)
810  FORMAT(10X,'>>>>OUT FLOW HYDROGRAPH FOR RESERVOIR ROUTING<<<<',/,
110X,'      TIME HRS                      WATER LEVEL (M)      ',/)
      TIME=0.
      DO 811 I=1,KENDT
      TIME=TIME+DT/3600.
      WRITE(14,820)TIME,STAGE(I)
      WRITE(*,820)TIME,STAGE(I)
820  FORMAT(14X,F5.2,25X,F6.2)
811  CONTINUE
C
C      COMPUTE MAXIMUM FLOOD (cms) TO DESIGN CHUTE SPILLWAY
C
      QPEAK=1.73*WL*((HWL-SPILL)**1.5)
      WRITE(9,621)QPEAK,WL
621  FORMAT(2F10.2)
      CLOSE(4)
      CLOSE(7)
      CLOSE(8)
      CLOSE(9)
      RETURN
      END
$DEBUG
      SUBROUTINE FLOW
C
C      SEEPAGE THROUGH EARTH DAM
C      BY FINITE DIFFERENCE METHOD
C
      IMPLICIT REAL(K)
      DIMENSION ID(500,4),F(500),H(500),A1(500),A2(500),AO(500),
1K(500,4),KK(500)
      COMMON/HYDRO7/ID,F,H,A1,A2,AO,K,KK
      INTEGER ALPHA,BETA
C      INPUT DATA
C      NB= NUMBER OF BOUNDARY NODES
C      N = NUMBER OF TOTAL NODES
C      ERR= ERROR CHECKED VALUES
      OPEN(6,FILE='FLOW.PRN')
      OPEN(16,FILE='FLOW.RES')
      READ(6,10)NB,N,ERR
10  FORMAT(2I8,F8.6)
C      INPUT DATA FOR BOUNDARY NODES
      DO 100 I=1,NB

```

```

WRITE(*,20)I
20 FORMAT(10X,'DATA AT NODE NO ',I4)
READ(6,30)ID(I,1),ID(I,2),ALPHA,BETA,XN,YN,DP,DI,F(I)
WRITE(*,30)ID(I,1),ID(I,2),ALPHA,BETA,XN,YN,DP,DI,F(I)
30 FORMAT(4I8,2F8.3,3F8.2)
A=(ALPHA+DP*BETA)/DI
A1(I)=A*YN
A2(I)=A*XN
AO(I)=A1(I)+A2(I)+BETA
H(I)=0
100 CONTINUE
C INPUT DATA FOR INTERIOR NODES
N1=NB+1
DO 200 I=N1,N
WRITE(*,110)I
110 FORMAT(10X,'DATA AT NODES NO.',I4)
READ(6,120)KK(I),ID(I,1),ID(I,2),ID(I,3),ID(I,4),
1K(I,1),K(I,2),K(I,3),K(I,4)
WRITE(*,120)KK(I),ID(I,1),ID(I,2),ID(I,3),ID(I,4),
1K(I,1),K(I,2),K(I,3),K(I,4)
120 FORMAT(F8.4,4I8,4F8.4)
H(I)=0
200 CONTINUE
50 ERRC=0
DO 300 I=1,NB
I1=ID(I,1)
I2=ID(I,2)
A=(F(I)+A1(I)*H(I1)+A2(I)*H(I2))/AO(I)
ERRC=ERRC+(A-H(I))**2
H(I)=A
300 CONTINUE
DO 400 I=N1,N
I1=ID(I,1)
I2=ID(I,2)
I3=ID(I,3)
I4=ID(I,4)
K1=K(I,1)
K2=K(I,2)
K3=K(I,3)
K4=K(I,4)
KO=KK(I)
A=((KO+K1)*H(I1)+(KO+K2)*H(I2)+(KO+K3)*H(I3)+(KO+K4)*H(I4))
A=A/(4*KO+K1+K2+K3+K4)
ERRC=ERRC+(A-H(I))**2
H(I)=A
400 CONTINUE
C COMPAIR ERROR
ERRC=SQRT(ERRC)
IF(ERRC.GT.ERR) GO TO 50
WRITE(16,1010)
WRITE(*,500)
500 FORMAT(10X,'PRESSURE HEAD AT NODES')
WRITE(16,500)
WRITE(*,1000)
WRITE(16,1000)

```

```

1000 FORMAT(10X,'  NODE                      H  ')
      WRITE(*,1010)
1010 FORMAT(10X,'*****')
      WRITE(16,1010)
      DO 600 I=1,N
        WRITE(*,1020)I,H(I)
        WRITE(16,1020)I,H(I)
1020 FORMAT(10X,I4,15X,F6.2)
600 CONTINUE
      CLOSE(6)
      CLOSE(16)
      RETURN
      END
$DEBUG
      SUBROUTINE LST(X1,Y1,N,M,A1)
C
C   METHOD OF LEAST SQUARE FOR CURVE FITTING
C   M=NUMBER OF PLOTTED POINTS
C   N=DEGREE OF PROPOSED POLYNOMIAL
C   MAXIMUM NUMBER OF DATA POINT = 20
C   DEGREE OF HIGHEST POLYNOMIAL = 3
C
      DIMENSION X1(20),Y1(20),XC(10),YX(5),A1(5),B1(5),C1(5,5),D1(5,5)
      NP1=N+1
      MP1=M+1
      M2=M*2
      DO 13 I=1,NP1
        WRITE(*,14)X1(I),Y1(I)
14    FORMAT(2F12.3)
13    CONTINUE
C    FORM SUMS OF POWERS OF X1(I)
      DO 20 K=1,M2
        XC(K)=0.0
        DO 15 I=1,NP1
15      XC(K)=XC(K)+X1(I)**K
20    CONTINUE
C    FORM SUM OF Y1(I)
      YC=0.0
      DO 22 I=1,NP1
22    YC=YC+Y1(I)
C    FORM SUMS OF PRODUCTS Y1(I)*X1(I)**K
      DO 30 K=1,M
        YX(K)=0.0
        DO 25 I=1,NP1
25      YX(K)=YX(K)+Y1(I)*X1(I)**K
30    CONTINUE
C    GENERATE NORMAL MATRIX C USING SUMS OF POWERS OF X(I)
      DO 36 I=1,MP1
        DO 35 J=1,MP1
          IPJM2=I+J-2
          IF(IPJM2)33,31,33
31      C1(1,1)=FLOAT(NP1)
          GO TO 35
33      C1(I,J)=XC(IPJM2)
35    CONTINUE

```

```

36 CONTINUE
C   GENERATE RIGHT-SIDE MATRIX B
    B1(1)=YC
    DO 37 I=2,MP1
37  B1(I)=YX(I-1)
C   INVERT NORMAL MATRIX C1
    CALL INVERSE (C1,D1,MP1)
C   MATRIX D1 IS INVERSE OF NORMAL MATRIX
C   FORM MATRIX PRODUCT OF INVERSE AND RIGHT-SIDE MATRIX
    DO 55 I=1,MP1
    A1(I)=0.0
    DO 54 J=1,MP1
54  A1(I)=A1(I)+D1(I,J)*B1(J)
55  CONTINUE
    RETURN
    END

$DEBUG
    SUBROUTINE INVERSE (A1,B1,N)
C   MATRIX INVERSION BY ELIMINATION WITH PARTIAL PIVOTING
C   ORIGINAL MATRIX = A1 INVERSE MATRIX =B1
    DIMENSION A1(5,5),B1(5,5)
    EPS = 0.0000001
C   CONSTRUCT IDENTITY MATRIX B1(I,J) =I
    DO 6 I=1,N
    DO 5 J=1,N
    IF (I-J)4,3,4
3   B1(I,J)=1.0
    GO TO 5
4   B1(I,J)=0.0
5   CONTINUE
6   CONTINUE
C   LOCATE MAXIMUM MAGNITUDE A1(I,K) ON OR BELOW MAIN DIAGONAL
    DEL=1.0
    DO 45 K=1,N
    IF(K-N)12,30,30
12  IMAX=K
    AMAX=ABS(A1(K,K))
    KP1=K+1
    DO 20 I=KP1,N
    IF(AMAX-ABS(A1(I,K)))15,20,20
15  IMAX=I
    AMAX=ABS (A1(I,K))
20  CONTINUE
C   INTERCHANGE ROWS IMAX AND K IF IMAX NOT EQUAL TO K
    IF(IMAX-K)25,30,25
25  DO 29 J=1,N
    ATMP=A1(IMAX,J)
    A1(IMAX,J)=A1(K,J)
    A1(K,J)=ATMP
    BTMP=B1(IMAX,J)
    B1(IMAX,J)=B1(K,J)
29  B1(K,J)=BTMP
    DEL=-DEL
30  CONTINUE
C   TEST FOR SINGULAR MATRIX

```

```
      IF (ABS(A1(K,K))-EPS)93,93,35
35  DEL=A1(K,K)*DEL
C   DIVIDE PIVOT ROW BY ITS MAIN DIAGONAL ELEMENT
      DIV=A1(K,K)
      DO 38 J=1,N
      A1(K,J)=A1(K,J)/DIV
38  B1(K,J)=B1(K,J)/DIV
C   REPLACE IS ROW BY LINEAR COMBINATION WITH PIVOT ROW
      DO 43 I=1,N
      AMULT=A1(I,K)
      IF(I-K)39,43,39
39  DO 42 J=1,N
      A1(I,J)=A1(I,J)-AMULT*A1(K,J)
42  B1(I,J)=B1(I,J)-AMULT*B1(K,J)
43  CONTINUE
45  CONTINUE
93  RETURN
      END
```