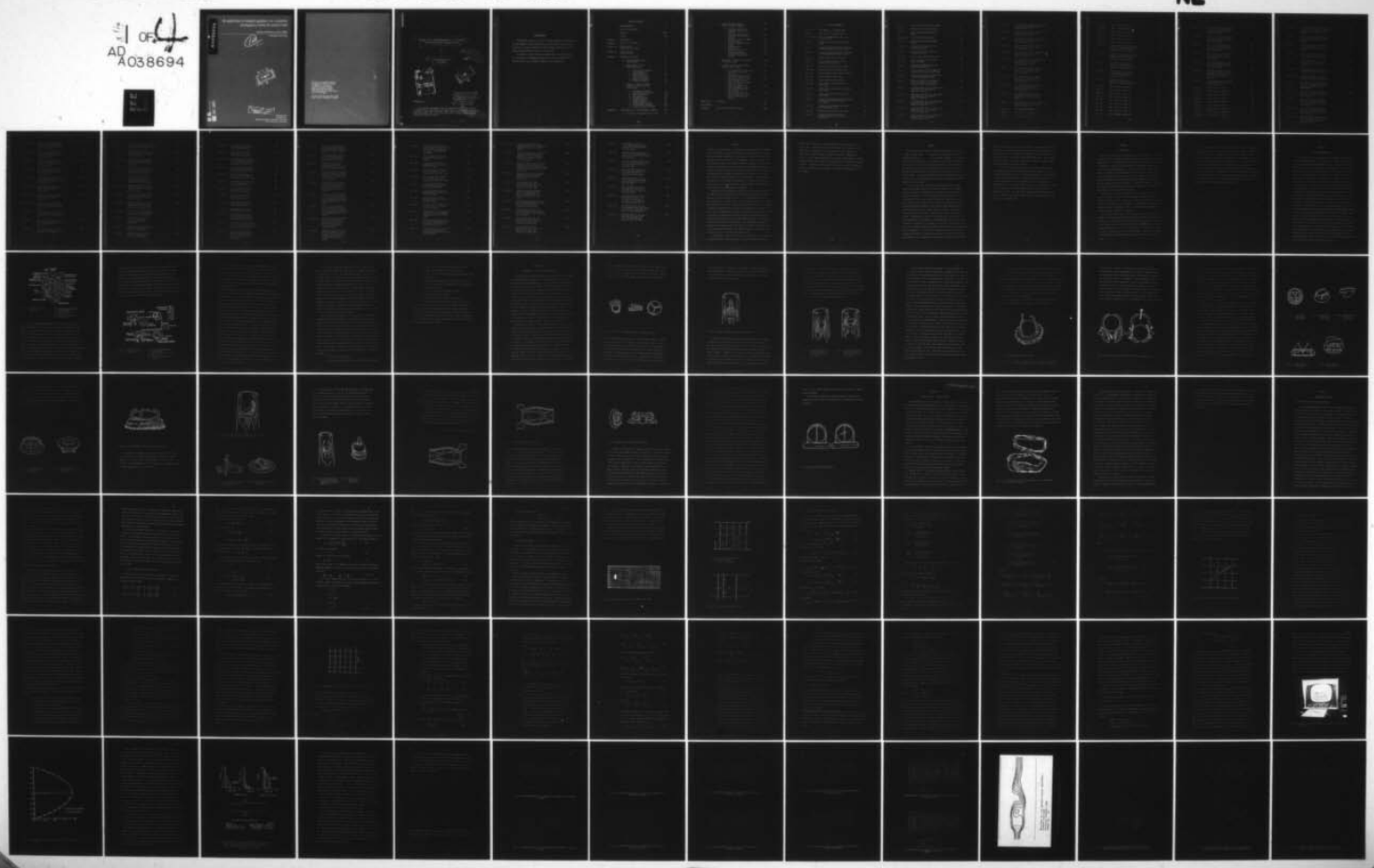
AD-A038 694 UTAH UNIV SALT LAKE CITY DEPT OF COMPUTER SCIENCE F/G 6/5
AN APPLICATION OF COMPUTEF GRAPHICS: TWO CONCURRENT INVESTIGATI--ETC(U)
NOV 71 H GREENFIELD, R DEBRY F30602-70-C-0300
UNCLASSIFIED UTEC-CSC-71-115 NL

1 OF 4
AD A038694



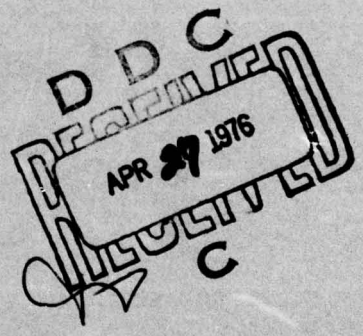
AD A038694

An application of computer graphics: two concurrent investigations within the medical field

HARVEY GREENFIELD, ROGER DEBRY

UNIVERSITY OF UTAH

12 NW



AD No.
DDC FILE COPY

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

NOVEMBER 1971
UTEC-CSc-71-115
COMPUTER SCIENCE, UNIVERSITY OF UTAH
SALT LAKE CITY, UTAH 84112

The views and conclusions contained in this document are those of the author(s) and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency of the U. S. Government.

This document has been approved for public release and sale; its distribution is unlimited.

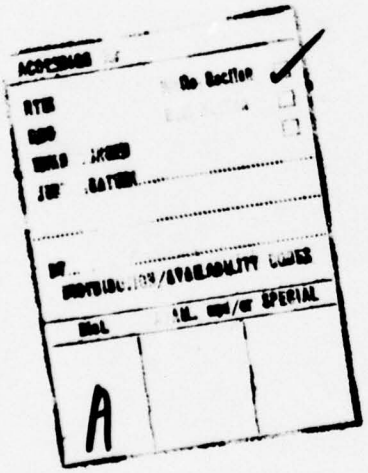
(6) AN APPLICATION OF COMPUTER GRAPHICS: TWO CONCURRENT INVESTIGATIONS WITHIN THE MEDICAL FIELD

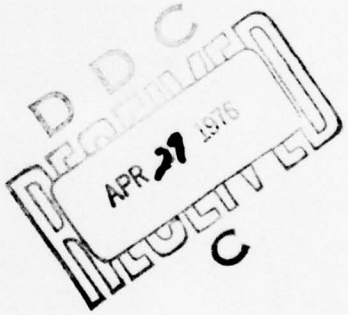
by

(11) Nov 71

(12) 299 p.

(10) Harvey Greenfield
Roger DeBry





DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

November 1971

(14) UTEC-CSc-71-115

(15) This research was supported by the Advanced Research Projects Agency of the Department of Defense, monitored by Rome Air Development Center, Griffiss Air Force Base, New York, 13440, under contract F30602-70-C-0300, ARPA order number 829. It was also supported by National Institutes of Health grant number HE-12202.

✓ PHS-HE-12202

404 949

ACKNOWLEDGMENT

Appreciation is due to Dr. David C. Evans, director of the University of Utah Computer Science Division and Dr. Willem Kolff, head, Division of Artificial Organs, University of Utah College of Medicine, for their unfailing and continuing support of this project.

Recognition is given, as well as appreciation, to Charles Brauer who initiated the programming responsible for the two dimensional flow studies while he was a graduate student at this institution.

TABLE OF CONTENTS

	Acknowledgements	ii
	List of Illustrations	v
	Abstract	xviii
	Preface	xx
CHAPTER I.	Introduction	1
CHAPTER II.	Heart Valve Mechanics	3
CHAPTER III.	Problem Area I: Artificial Heart Valve	9
CHAPTER IV.	Problem Area II: Atherosclerosis	27
CHAPTER V.	Mathematical Approach	31
	A. Some Preliminary Fluid Flow Concepts	31
	B. Determination of Flow About an Obstacle	32
	1. Navier-Stokes Equations Forms Required	33
	2. Computational Mesh	37
	3. Finite Difference Equations Required	40
	4. Numerical Procedure	45
	5. Boundary Conditions	54
	C. Display of Computer Graphics Results - Obstacle and Mesh Coincidental	57
	1. Rectangular Obstacle	65
	2. Semi-hemisphere Obstacle	75
	3. Spherical Obstacle	84
	4. Oval Obstacle	95
	5. Octagonal Obstacle	105
	6. Angled Simplified Leaflet Obstacle	116
	7. Simple Entrance Regions	124
	8. Two Arbitrary Obstacles	136
	9. Simple Aircraft Wing Shape	143
CHAPTER VI.	Presentation of a Curved Boundary Algorithm	145
	A. Discussion for Atherosclerosis Study	150

B. Curved Boundary Computer Graphic Displays (Vascular) 154

1. Idealized, Orthogonal Branching, Renal Artery Section 155
2. Idealized, Non-orthogonal Branching, Renal Artery Section 161
3. Montage of Portion of Renal Artery Section 167
4. Idealized Atherosclerotic Plaque 168
5. Abdominal Aortic Bifurcation 169
6. Arterial "Tree" with sites for Plaques 175
7. Renal-Abdominal Aortic Bifurcation Combination Section 176

C. Discussion of Concern to Prosthetic Heart Valve Study 183

D. Curved Boundary Computer Graphic Displays (Heart Valve) 185

1. Starr-Edwards Type Artificial Heart Valve; Full Open Position 186
2. Disc-Type Artificial Heart Valve; Full Open Position 192
3. Flow Pattern About a Disc-Type Prosthetic Heart Valve in a Mock Circulation (One Time Frame) 198
4. Vorticity Function Plot for Starr-Edwards Heart Valve Type at Partially Open Position 199
5. Vorticity Function Plot for Disc-Type Heart Valve Type at Partially Open Position. 200

CHAPTER VII. Conclusions 201

BIBLIOGRAPHY 205

APPENDIX: Flow Chart and Computer Program 211

LIST OF ILLUSTRATIONS

Fig. 1:	The Chambers of the Human Heart.	4
Fig. 2:	The Valves of the Human Heart.	5
Fig. 3,a,b,c:	Ball, Disc(Hammersmith form), and Leaflet Valves.	10
Fig. 4:	Flexible Cusp Type Artificial Heart Valves.	11
Fig. 5:	Diagram Showing Flow Direction Through a Ball-Type Artificial Heart Valve.	12
Fig. 6:	Diagram Showing Flow Direction Through a Disc-Type Artificial Heart Valve.	12
Fig. 7:	MaGovern Artificial Heart Valve.	14
Fig. 8:	Smeloff-Cutter Full Flow Orifice Artificial Heart Valve.	15
Fig. 9:	Cross-Jones Artificial Heart Valve.	17
Fig. 10:	Kay-Shiley Artificial Heart Valve.	17
Fig. 11:	Melrose-Alvarez Artificial Heart Valve.	17
Fig. 12:	Gott Artificial Heart Valve.	17
Fig. 13:	Davila Artificial Heart Valve.	17
Fig. 14:	Starr-Edwards Disc-Type Artificial Heart Valve.	18
Fig. 15:	Harken Mitral Discoid-Type Artificial Heart Valve.	18
Fig. 16:	Cooley-Bloodwell-Cutter Artificial Heart Valve.	19
Fig. 17:	Diagram Showing Flow Direction Through a Hinged Leaflet-Type Artificial Heart Valve.	20
Fig. 18:	Lillehei-Kaster Pivoting Disc-Type Artificial Heart Valve.	20
Fig. 19:	Wada-Cutter Artificial Heart Valve.	20
Fig. 20:	Diagram Shwoing Flow Direction Through A Poppet-Type Artificial Heart Valve.	21

Fig. 21:	Barnard UCT Artificial Heart Valve.	21
Fig. 22:	Vorhauer Artificial Heart Valve.	22
Fig. 23:	Nakib Toroidal Artificial Heart Valve.	23
Fig. 24,a.b:	Lethal Mitral Valve Variance.	25
Fig. 25:	Diseased Atherosclerotic Human Artery.	28
Fig. 26:	An Idealized Obstacle and Its Computational Mesh.	38
Fig. 27:	Enlarged Portion of the Basic Computational Mesh.	39
Fig. 28:	The Irregular Computational Mesh.	39
Fig. 29:	Finite Difference Stencil Near A Curved Boundary.	44
Fig. 30:	Computational Mesh Form for the Pearson-Esch Approach.	49
Fig. 31:	A Typical Result for Flow About an Obstacle As Seen on the IDI Scope.	59
Fig. 32:	Comparison of Computer Results with Predicted Parabolic Profile.	60
Fig. 33:	Stability Values of the Computer Numerical Routine for Certain Stream and Vorticity Functions.	62
Fig. 34:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 23).	65
Fig. 35:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 26).	65
Fig. 36:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 29).	66
Fig. 37:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 32).	66
Fig. 38:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 35).	67

Fig. 39:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 38).	67
Fig. 40:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 41).	68
Fig. 41:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 44).	68
Fig. 42:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 47).	69
Fig. 43:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle (Frame 50).	69
Fig. 44:	Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle and For A Higher Reynolds Number Value.	70
Fig. 45:	The Horizontal Velocity Function Plot for a Rectangular Obstacle (Surface and Isometric Views At One Time Frame).	71
Fig. 46:	The Vertical Velocity Function Plot for a Rectangular Obstacle (Surface and Isometric Views at One Time Frame).	72
Fig. 47:	The Total Velocity Function Plot For a Rectangular Obstacle (Surface and Isometric Views at One Time Frame).	73
Fig. 48:	The Velocity Vector Plot for a Rectangular Obstacle (Surface and Isometric Views at One Time Frame).	74
Fig. 49:	Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle (Frame 23).	75
Fig. 50:	Same as above at Frame 25.	75
Fig. 51:	Same as above at Frame 29.	76
Fig. 52:	Same as above at Frame 32.	76

Fig. 53:	Same as above at Frame 35.	77
Fig. 54:	Same as above at Frame 38.	77
Fig. 55:	Same as above at Frame 41.	78
Fig. 56:	Same as above at Frame 44.	78
Fig. 57:	Same as above at Frame 47.	79
Fig. 58:	Same as above at Frame 50.	79
Fig. 59:	The Horizontal Velocity Function Plot for a Semi-Hemisphere Obstacle (Surface and Isometric Views at One Time Frame).	80
Fig. 60:	The Vertical Velocity Function Plot for a Semi-Hemisphere Obstacle (Surface and Isometric Views at One Time Frame).	81
Fig. 61:	The Total Velocity Function Plot for a Semi-Hemisphere Obstacle (Surface and Isometric Views at One Time Frame).	82
Fig. 62:	The Velocity Vector Plot for a Semi-Hemisphere Obstacle (Surface and Isometric Views at One Time Frame).	83
Fig. 63:	Stream Function Plot at a Particular Time Frame for a Spherical Obstacle (Frame 23).	84
Fig. 64:	Same as above at Frame 25.	84
Fig. 65:	Same as above at Frame 29.	85
Fig. 66:	Same as above at Frame 32.	85
Fig. 67:	Same as above at Frame 35.	86
Fig. 68:	Same as above at Frame 38.	86
Fig. 69:	Same as above at Frame 41.	87
Fig. 70:	Same as above at Frame 44.	87
Fig. 71:	Same as above at Frame 47.	88
Fig. 72:	Same as above at Frame 50.	88

Fig. 73:	The Horizontal Velocity Function Plot for a Spherical Obstacle (Surface and Isometric Views at One Time Frame).	89
Fig. 74:	The Vertical Velocity Function Plot for a Spherical Obstacle (Surface and Isometric Views at One Time Frame).	90
Fig. 75:	The Total Velocity Function Plot for a Spherical Obstacle (Surface and Isometric Views at One time Frame).	91
Fig. 76:	The Velocity Vector Plot for a Spherical Obstacle (Surface and Isometric Views at One Time Frame).	92
Fig. 77:	An Experimental View of Flow Streamlines About an Obstacle as Formed by Smoke Injection (0.5 in. cylinder).	93
Fig. 78:	Experimental Views of Flow Streamlines About an Obstacle as Formed by Smoke Injection (At Increasing Reynolds Numbers).	94
Fig. 79:	Stream Function Plot at a Particular Time Frame for An Oval Obstacle (Frame 23).	95
Fig. 80:	Same as above at Frame 25.	95
Fig. 81:	Same as above at Frame 29.	96
Fig. 82:	Same as above at Frame 32.	96
Fig. 83:	Same as above at Frame 35.	97
Fig. 84:	Same as above at Frame 38.	97
Fig. 85:	Same as above at Frame 41.	98
Fig. 86:	Same as above at Frame 44.	98
Fig. 87:	Same as above at Frame 47.	99
Fig. 88:	Same as above at Frame 50.	99

Fig. 89:	The Horizontal Velocity Function Plot for an Oval Obstacle (Surface and Isometric Views at One Time Frame).	100
Fig. 90:	The Vertical Velocity Function Plot for an Oval Obstacle (Surface and Isometric Views at One Time Frame).	101
Fig. 91:	The Total Velocity Function Plot for an Oval Obstacle (Surface and Isometric Views at One Time Frame).	102
Fig. 92:	The Velocity Vector Plot for an Oval Obstacle (Surface and Isometric Views at One Time Frame).	103
Fig. 93:	A Stream Function Plot, at a Particular Time Frame and High Reynolds Number for an Oval Obstacle.	104
Fig. 94:	The Stream Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 10).	105
Fig. 95:	The Stream Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 25).	106
Fig. 96:	The Horizontal Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 10).	107
Fig. 97:	The Horizontal Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 25).	108
Fig. 98:	The Vertical Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 10).	109
Fig. 99:	The Vertical Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 25).	110
Fig. 100:	The Total Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 10).	111

Fig. 101:	The Total Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 25).	112
Fig. 102:	The Vorticity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 10).	113
Fig. 103:	The Vorticity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at Time Frame 25).	114
Fig. 104:	The Velocity Vector Plot for an Octagonal Obstacle (surface and Isometric Views at Time Frame 25).	115
Fig. 105:	Stream Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet; Reynolds number = 200. (One Time Frame).	116
Fig. 106:	Stream Function Plot for Angled One-Cusp Rigid Heart Valve Leaflets; Reynolds number = 900.	117
Fig. 107:	Vorticity Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet; Reynolds number = 200.	118
Fig. 108:	Horizontal Velocity Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet; Reynolds number = 900.	119
Fig. 109:	Vertical Velocity Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet; Reynolds number = 900.	120
Fig. 110:	Total Velocity Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet; Reynolds number = 900.	121
Fig. 111:	Velocity Vector Plot for Angled One-Cusp Rigid Heart Valve Leaflet; Reynolds number = 200.	122

Fig. 112:	Velocity Vector Plot for Angled One-Cusp Rigid Heart Valve Leaflet; Reynolds number = 900.	123
Fig. 113:	Stream Function Plot for Idealized Flow into Simplified Atrium Section of the Heart. (One Time Frame).	124
Fig. 114:	Horizontal Velocity Function Plot for Idealized Flow into Simplified Atrium Section of the Heart.	125
Fig. 115:	Vertical Velocity Function Plot for Idealized Flow into Simplified Atrium Section of the Heart; One Time Frame.	126
Fig. 116:	Total Velocity Function Plot for Idealized Flow into Simplified Atrium Section of the Heart; One Time Frame.	127
Fig. 117:	Vorticity Function Plot for Idealized Flow into Simplified Atrium Section of the Heart; One Time Frame.	128
Fig. 118:	Velocity Vector Plot for Idealized Flow into Simplified Atrium Section of the Heart; One Time Frame.	129
Fig. 119:	Stream Function Plot for Two Arbitrary Shaped Flow Areas (Amenable to Particular Applications); One Time Frame.	130
Fig. 120:	Horizontal Velocity Function Plot for Two Arbitrary Shaped Flow Areas; One Time Frame.	131
Fig. 121:	Vertical Velocity Function Plot for Two Arbitrary Shaped Flow Areas; One Time Frame.	132
Fig. 122:	Total Velocity Function Plot for Two Arbitrary Shaped Flow Areas; One Time Frame.	133
Fig. 123:	Vorticity Function Plot for Two Arbitrary Shaped Flow Areas; One Time Frame.	134

Fig. 124:	Velocity Vector Plot for Two Arbitrary Shaped Flow Areas; One Time Frame.	135
Fig. 125:	Stream Function Plot for Two Obstacles in a Flow Field; One Time Frame.	136
Fig. 126:	Photograph from IDI Console Screen Showing Stream Function Plotted for Detailed Study of Internal Core of Flow.	137
Fig. 127:	Horizontal Velocity Function Plot for Two Obstacles in a Flow Field; One Time Frame.	138
Fig. 128:	Vertical Velocity Function Plot for Two Obstacles in a Flow Field; One Time Frame.	139
Fig. 129:	Total Velocity Function Plot for Two Obstacles in a Flow Field; One Time Frame.	140
Fig. 130:	Vorticity Function Plot for Two Obstacles in a Flow Field; One Time Frame.	141
Fig. 131:	Velocity Vector Plot for Two Obstacles in a Flow Field; One Time Frame.	142
Fig. 132:	Stream Function Plot for an Arbitrary Aircraft Wing Shape; One Time Frame.	143
Fig. 133:	Computational Mesh With Curved Boundary Involved.	146
Fig. 134:	Stream Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries); One Time Frame.	155
Fig. 135:	Horizontal Velocity Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries); One Time Frame.	156
Fig. 136:	Vertical Velocity Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries); One Time Frame.	157

Fig. 137:	Total Velocity Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries); One Time Frame.	158
Fig. 138:	Vorticity Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries); One Time Frame.	159
Fig. 139:	Velocity Vector Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries); One Time Frame.	160
Fig. 140:	Stream Function Plot for Non-Orthogonal Idealized Arterial Branching (Both Facing Downstream), At the Renal Section; One Time Frame.	161
Fig. 141:	Stream Function Plot for Branching Arteries at Renal Section; One Branch Facing Upstream and One Branch Facing Downstream (One Time Frame).	162
Fig. 142:	Horizontal Velocity Function Plot for Branching Arteries at Renal Section; One Branch Facing Upstream and One Branch Facing Downstream (One Time Frame).	163
Fig. 143:	Vertical Velocity Function Plot for Branching Arteries at Renal Section; One Branch Facing Upstream and One Branch Facing Downstream (One Time Frame).	164
Fig. 144:	Total Velocity Function Plot for Branching Arteries at Renal Section; One Branch Facing Upstream and One Branch Facing Downstream (One Time Frame).	165
Fig. 145:	Vorticity Function Plot for Branching Arteries at Renal Section; One Branch Facing Upstream and One Branch Facing Downstream (One Time Frame).	166

Fig. 146:	Montage of Portion of Renal Section with Side Artery Facing Upstream. At Different Flow Velocities, Showing Vortex Development.	167
Fig. 147:	Various Function Plots for an Idealized Atherosclerotic Plaque.	167
Fig. 148:	Stream Function Plot for Abdominal Aortic Bifurcation, (One Time Frame).	169
Fig. 149:	Horizontal Velocity Function Plot for Abdominal Aortic Bifurcation, (One Time Frame).	170
Fig. 150:	Vertical Velocity Function Plot for Abdominal Aortic Bifurcation, (One Time Frame).	171
Fig. 151:	Total Velocity Function Plot for Abdominal Aortic Bifurcation, (One Time Frame).	172
Fig. 152:	Vorticity Function Plot for Abdominal Aortic Bifurcation, (One Time Frame).	173
Fig. 153:	Velocity Vector Plot for Abdominal Aortic Bifurcation, (One Time Frame).	174
Fig. 154:	Arterial "Tree" Showing Sites of Predilection for Atherosclerotic Plaques.	175
Fig. 155:	Stream Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; One Time Frame.	176
Fig. 156:	Horizontal Velocity Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; One Time Frame.	177
Fig. 157:	Vertical Velocity Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; One Time Frame.	178

Fig. 158:	Total Velocity Function for Renal-Abdominal Aortic Bifurcation Combination Section; One Time Frame.	179
Fig. 159:	Vorticity Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; Orthogonal Branching at Renal Position, (One Time Frame).	180
Fig. 160:	Vorticity Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; Non-Orthogonal Branching at Renal Position, (One Time Frame).	181
Fig. 161:	Velocity Vector Plot for Renal-Abdominal Aortic Bifurcation Combination Section; (One Time Frame).	182
Fig. 162:	Stream Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position; One Time Frame.	186
Fig. 163:	Horizontal Velocity Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position; One Time Frame.	187
Fig. 164:	Vertical Velocity Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position; One Time Frame.	188
Fig. 165:	Total Velocity Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position; One Time Frame.	189
Fig. 166:	Vorticity Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position; One Time Frame.	190
Fig. 167:	Velocity Vector Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position; One Time Frame.	191

Fig. 168:	Stream Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position; One Time Frame.	192
Fig. 169:	Horizontal Velocity Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position; One Time Frame.	193
Fig. 170:	Vertical Velocity Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position; One Time Frame.	194
Fig. 171:	Total Velocity Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position; One Time Frame.	195
Fig. 172:	Vorticity Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position; One Time Frame.	196
Fig. 173:	Velocity Vector Plot for a Disc-Type Artificial Heart Valve at Full Open Position; One Time Frame.	197
Fig. 174:	Flow Pattern About a Disc-Type Artificial Heart Valve in a Mock Circulation, at One Time Point (courtesy of D. Wieting, Baylor Univ.).	198
Fig. 175:	Vorticity Function Plot for a Starr-Edwards Type Artificial Heart Valve at Partially Open Position (One Time Frame).	199
Fig. 176:	Vorticity Function Plot for a Disc-Type Artificial Heart Valve at Partially Open Position (One Time Frame).	200

ABSTRACT

The aim of the project is to apply new or recently developed computer graphic techniques to a particular discipline with the thought of broadening its research capabilities. The discipline chosen, that of medicine, has emphasis placed upon the area of hemodynamics. It is seen that computer graphics methods lend themselves admirably to studies of blood flow subsequent to analyses of particular problems by the application of fluid mechanics concepts and numerical analysis techniques. Projects chosen for contemplation were the design of optimized flow prosthetic heart valves and the study of vascular degeneration and replacement. A developed mock circulation is to be employed concurrently for furnishing verification of the theoretical results, a relatively uncommon attempt to experimentally verify particular computer graphics algorithms.

Prosthetic heart valves may be classified for fluid flow operational purposes according to the presence of an unobstructed central fluid pathway upon valve opening (leaflet type), or a central obstruction with lateral flow (ball or poppet type). In all situations, blood as the fluid, sees obstacles in its path of flow and the project, therefore, is to attempt to duplicate mathematically this flow of fluid about such an obstacle. In turn, if an idealized obstacle shape is transferred from the center of flow to one wall, the mathematical problem is basically the same but now an idealized atherosclerotic lesion can be formed for study. In continuing to show the versatility of the project, if one follows the blood flow through the circulatory system, the corners of arterial side channels constitute mathematical obstacles since abrupt changes in the flow direction are caused, as well as distortion of the flow lines.

In attempting to apply viscous fluid mechanics theory one is required to employ the Navier-Stokes equations, a partial differential equation

system which is nonlinear, thus allowing exact solutions in only a few special cases. These equations are within major areas of application of modern applied mathematics since they exhibit all the computational difficulties of other partial differential equations of mathematical physics. Direct numerical solution of the Navier-Stokes equations, however, now seems within reach because of the availability of large memory digital computers. Subsequently, the application of physical laws to hemodynamic studies is enhanced by the attachment of a graphical display system to the computer.

PREFACE

The authors of this report are dedicated to the task of effecting a useful and true man-machine interface by the expanded utilization of computer graphics as a complement to the digital computer. Their own area of interest is centered about the development of interactive techniques for the solution of large-scale numerical problems. Such problems, at the present, are difficult, if not impossible to solve practically by machine computing processes and require the guidance and direction of a skilled human. In particular, graphically-aided techniques for the practical solution of nonlinear partial differential equations are of prime importance in this study.

It is the contention of the present investigators that a large percentage, perhaps nearly all, of algorithms developed for computer graphics usage are the property of a relatively small elite group of computer scientists. If so, then certainly the potentialities of such algorithms are lost to many other disciplines for application. With this in mind, it was decided that one of the more conservative of other disciplines should be broached in an attempt to make them aware of the innate capabilities of the computer graphics concept. The discipline of medicine was so chosen since they are particularly sensitive to the inroads of mathematical concepts. The medical world is newly aware of the contributions of the computer to the measurement, monitoring and data collection of concern to the treatment of critically ill patients, however, it is relatively unaware of computer graphics usage for understanding medical phenomenon. To this end, two medical investigations have been pursued; the design of an optimized flow artificial heart valve and the understanding of the formation and positioning of atherosclerotic lesions.

Bulk of the funding for such applied studies is provided by the National Institute of Health under grant HE 12201, with Advanced Research Project Agency support for the interactive techniques on the area of basic algorithms useful for application to other ARPA study areas. In the large, it is an attempt to make other disciplines cognizant of computer aided design and graphics usage more catholic.

The first of these reports discusses the numerical and resulting algorithms required for calculating the transient dynamics of a specific incompressible fluid problem. The approach developed the two-dimensional results and allowed formation of resulting configurations with the aid of Univac 1108 and PDP-8 digital computers plus an IDI display and Gerber plotter. The companion report continues the mathematical technique so as to satisfy free surface boundary conditions and aid in three-dimensional studies. Here the desired results were formed with the aid of PDP-9 and PDP-10 assemblies, plus a Univac 1559 display system. The project is a continuing one.

CHAPTER I

INTRODUCTION

The aim of the project that is presented in this report is to enlarge the usage of present computer graphic capabilities in a new area of endeavor, namely, medicine. The employment of fluid mechanics concepts is concurrent. In attempting to apply viscous fluid mechanics theory one is required to study the Navier-Stokes equations, a partial differential equation system which is nonlinear, thus allowing exact solutions in only a few special cases. These equations are within major areas of application of modern applied mathematics since they exhibit all the computational difficulties of other partial differential equations of mathematical physics. Algorithms and concepts formed within this investigation should then be capable of applicability in other investigative areas.

Although only specialized situations have been fully explored in the past, of concern to the Navier Stokes equations, it appears that direct numerical solution of such a system now is within reach because of the availability of large memory digital computers. Subsequently, the application of physical laws to hemodynamic studies is enhanced by the attachment of a graphical display system to the computer.

Two particular applications are presented in this study. One is an attempt to study blood flow through artificial heart valves and employ such information to eventually design an optimized flow prosthetic heart valve. The other application is the study of vascular degeneration or the attempt to amplify knowledge of concern to atherosclerosis. Background knowledge, of concern to both topics, are discussed somewhat in depth, for in appealing to interdisciplinary knowledge dispersion, such an action is thought necessary.

The applications appear, at first glance, to be quite dissimilar, however, in most artificial heart valves, blood as a fluid, sees obstacles in the flow path. The procedure, therefore, is an attempt to mathematically duplicate the flow of fluid about such an obstacle. In turn, if an idealized obstacle shape is transferred from the center of flow to one wall, the mathematical problem is basically the same but now an idealized atherosclerotic lesion is formed for study. In continuing to show the versatility of the approach, if one follows the blood flow through the circulatory system, the corners of arterial side channels constitute mathematical obstacles since abrupt changes in the flow direction are caused, as well as distortion of the flow lines.

CHAPTER II

HEART VALVE MECHANICS

The human heart, for all its fascinating capabilities, is basically just a four chambered pump. It is located in the lower central portion of the chest cavity, just behind the breast bone and between the right and left lungs. The male human heart is approximately four and one half inches long, three and one half inches wide at its widest point, and two and one half inches thick. The male human heart weighs from 10 to 12 ounces while the female heart weighs from 8 to 10 ounces. There is a partition that divides the heart into right half and left half sections. In turn, each half is divided into two more chambers which are termed the atrium and ventricle. The atria act as collecting chambers for blood returning to the heart from the lungs and the veins of the body. The atria fill the ventricles with blood before each beat when the ventricles are relaxed. The ventricles are muscular chambers which initiate the circulation of blood through portions of the body and ultimately back to the heart. The left ventricle is the true pumping muscle of the heart since it not only sends blood through the body but it must pump against gravity so as to forward blood to the head and other body sections above the heart region. The right ventricle needs only to furnish blood to the lungs. Each ventricle has an inflow and outflow valve. As shown in figure 1, the left ventricle's inflow valve is the mitral valve and the aortic valve is the outflow valve. The inlet valve for the right ventricle is the tricuspid valve while the outflow valve is the pulmonary valve. A complete discussion of the anatomy involved is not in order, however, the above short discussion introduces the reader to a brief description of the chambers that require valves for proper blood motion.

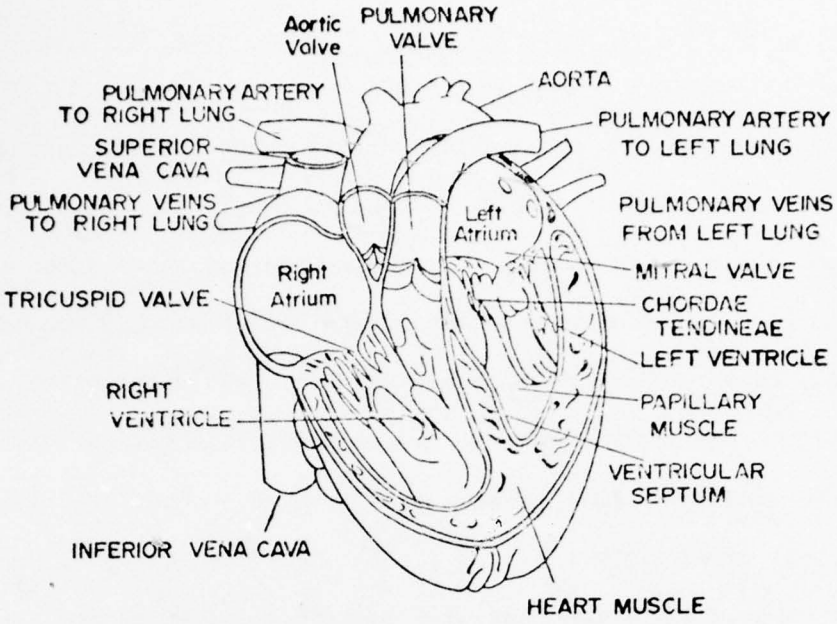


Fig. 1: The Chambers of the Human Heart

From Structure And Function Of The Heart
Copyright @ 1964 by D.C. Heath and Co.
Reprinted by Permission of Publishers.

In turn, to understand subsequent fluid dynamic and computer graphic requirements, the four heart valves will be discussed in gross detail. Between each atrium and ventricle there is an opening encapsulated by a fibrous ring. The valves are fibrous tissue in leaflet form which extend from the ring into the ventricle. Delicate cords run from the free edges of the leaflets to certain muscles of the ventricles. These cords form a restraining mechanism so that when the ventricle collapses, these leaflets close tight but do bend somewhat in a reverse fashion into the atrium and allow some leakage. The mitral valve, between the left atrium and ventricle has two leaflets that act as a one way valve. The two leaflets are uneven in size and one overlaps the other slightly.

In the case of the tricuspid valve, which is the one way valve between the right atrium and right ventricle, it consists of three leaflets or cusps. The pulmonary valve allows one way blood flow from the right ventricle to the pulmonary artery while the aortic valve allows one way blood flow from the left ventricle to the aorta. As seen in figure 2, both these valves have three pouchlike leaflet sections which, when opened, press against the arterial wall. When blood attempts a reverse flow the pouches immediately fill and the ensuing blood pressure forces closure of the three leaflets and prevents backflow.

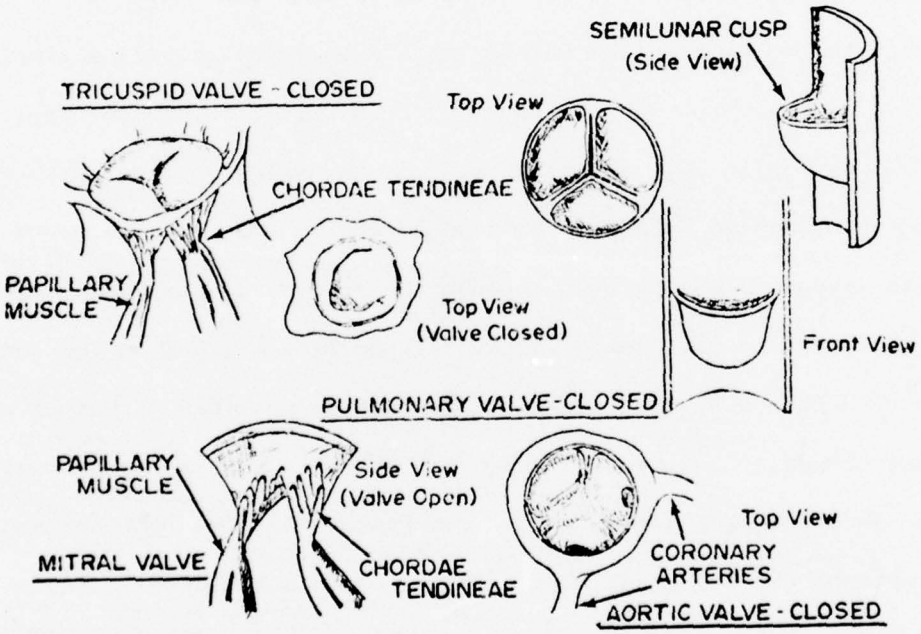


Fig. 2: The Valves of the Human Heart

From Structure And Function Of The Heart
Copyright @ 1964 by D.C. Heath and Co.
Reprinted by Permission of Publishers.

The above discussed heart valves, outstanding examples of nature's engineering capabilities that they are, may be altered or destroyed, however, by congenital malformation or by disease. Generally, it is the mitral or the aortic valve that is affected. There may be insufficiency (regurgitation) due to stiffening and shortening of the valve leaflets which prevents full closure, or there may be stenosis due to narrowing of the valve orifice.

A description of the complicated fluid flow phenomena occurring through the aortic valve, as an example, impresses one with the difficult problem proposed for the computer graphics investigation. Valvular and vascular flows encounter eddying, vortex flow, stagnation regions, jet impingement, both laminar and turbulent flow, and even cavitation. If one models the movement of an aortic valve even in relatively a simple fashion, it is seen that at low rates of pulsatile flow, a flow trail forms from the valve edge at either side of the valve orifice, with flow becoming turbulent downstream. With an advancing flow rate, a central turbulent stream is encountered through the open valve, extending into the aortic arch and descending aorta. If the flow narrows in the aorta because of a stenosis, then a jet is a resulting possible formation at the constriction. As flow rate again increases, a vortex forms where the jet edges meet the aortic wall. Continuing the flow increase naturally extends the jet downstream with increased turbulence aiding in changing the vortex direction in the ascending aorta. The center core of the jet strikes the opening of the innominate artery, allowing two flow branches to form with one compounding the vortex motion and the other continuing along the vascular wall. A stream of vortices forms along the aortic arch while eddying motions appear at branches downstream of the aorta. No viscous damping of turbulence occurs in the descending aorta since flow velocity remains high.

In vivo, these eddies that stream from the valve edges are functional in that they prevent the valve cusps from completely retracting, thus allowing the valve to close quickly when required, and to keep coronary vessels open. Eddy trails are simultaneously generated in the tri-cusp semilunar valve and are deflected in the aortic sinus and arch; each contributing to continued eddy and pulsatile flow. There is evidence that the core of the jet flow is deflected with a periodical lateral motion at the wall due to the wall's elasticity. The advent of helical flow motion is seen, generated by centrifugal force in the curved aortic arch which subsequently allows secondary flow, forcing turbulent action down to the origin of main vascular branches. Pulsatile flow with its inclusion of rapid minimum-maximum-minimum velocities incurs brief backflows. Turbulence and eddy formation from these sources all add to the description of a flow pattern.

The realization that surgery is not the answer to salvaging many irreparably damaged heart valves led investigators to the thought of replacement by artificial heart valves. With the advent of the artificial heart valve it is to be noted, however, that the many fluid dynamic problems projected in a cursory way in the above paragraph are still with us, and the resolution of such problems do not enter into the design of such an artificial device. New models are continually being introduced; even cadaver valves are placed in metal rings for insertion, yet dissatisfaction is still indicated with presently available valves. As Braunwald and Detmer [1] pointed out, no currently available artificial heart valve fulfills all the acceptance criteria for an ideal substitute, as proposed by Harken [2], i.e.,

1. It must not propagate emboli.
2. It must be chemically inert, and not damage blood elements.

3. It must offer no resistance to physiologic flows.
4. It must close promptly (less than 0.05 sec.)
5. It must remain closed during the appropriate phase of the cardiac cycle.
6. It must have lasting physical and geometric features.
7. It must be inserted in a physiologic site, generally the normal anatomic site.
8. It must be capable of permanent fixation.
9. It must not annoy the patient.
10. It must be technically practical to insert.

In fact, true functional evaluation and comparison of present artificial heart valves is rare among those who manufacture them, beyond the relatively unsophisticated quality control procedures. This is not to say, however, that in general, a substitute valve does not serve its purpose well. The patient has been given a considerably improved outlook over the obvious alternative, with even the presently available prosthetic heart valves.

CHAPTER III

PROBLEM AREA I: ARTIFICIAL HEART VALVES.

The first successful implantation of a prosthetic heart valve appears to have been performed by Hufnagel [3] who implanted a form of what has become known as the ball-valve type. The success was not spectacular since improvement to upper extremity arterial blood flow was not accomplished. This was due to the requirement of cross clamping of certain arteries, for the heart lung machine was not to be invented until the following year. It remained for Starr [4], however, to achieve impressive results after Harken [5] implanted a prosthetic valve with aid of the heart lung machine in 1960. Since then, some 60,000 artificial heart valves have been implanted in Americans with varying degrees of success. Subsequently, some 50 to 75 types of artificial heart valves have been devised, with the Starr-Edwards ball type valve being used most extensively. Still present are problems of thromboembolus, red cell destruction, material fatigue and chemical change, leakage, and infection.

Mention has been made of a "ball-type" prosthetic device. Basically, prosthetic heart valves may be classified as those with central flow or lateral flow. The natural leaflet valve, as shown in figure 3c, allows an unobstructed central pathway for blood flow as the leaflets open. On a cross-sectional view an artificial leaflet valve appears as shown in figure 4. In attempting to replace the natural valve investigators found that no man made material can withstand the constant flexing, therefore, heterograft (from other species) or homograft (from human cadavers) leaflet valves are sometimes employed. The attribute of a homograft valve is that there appears to be a lesser amount of thromboembolism occurrence than for a mechanical valve, however, a statistical

study of subsequent durability is still lacking. Aortic insufficiency, or poor blood flow when replacement is made for the aortic valve with homograft leaflets, is present due to the difficult operative technique required, plus insufficient methods of preservation for the leaflets.

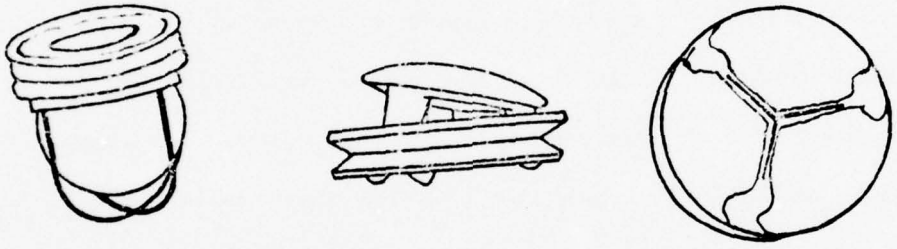


Fig. 3,a,b,c: Ball, Disc(Hammersmith) and Leaflet Valves.

There is some evidence by Malm et al. [6] that leaflets exposed to a high energy electron beam allows superior tensile strength, however, such a sterilization technique is not widely available nor may it be permanent. A problem relative to leaflet valve design is its pliability and the requirement of integrated motion of the leaflets with the bloodstream velocity for negating a shearing effect. Consequently, single leaflet or multiple cusp designs need to be precisely angled to the blood flow

velocity, and this is presently not attainable. Studies employing pulse duplicators show one leaflet of a tricuspid aortic valve flexing with the other leaflets allowing turbulence and cavitation to be initiated on the conforming side of the valve.

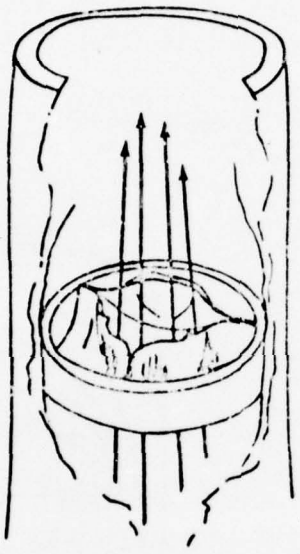


Fig. 4: Flexible Cusp Type Artificial Heart Valves

Viewing the lateral flow classification of prosthetic heart valves, as mentioned, the most popular has been the type with a caged ball, for the aortic position, as shown in figures 3a and 5, and as discussed later, the caged disk or hinged form for the mitral replacement, as shown in figures 3b and 6. (Although the same type of valve has been greatly used for all valvular positions, in different sizes, it appears that different blood flow regimes are present in the aortic and mitral positions. This comment requires further investigation). Interestingly enough, as late

as 1969, Duvoisin and M^{c}Goon [7] of the Mayo clinic mentioned that the transition to a ball type valve from the difficult to manufacture leaflet type valve was an easy decision to make "by the realization that for certain other mechanical functions found in nature, man has made practical improvements in design, such as the jet engine over the flapping wing of the bird, and the screw over the fish's tail". Rather an over-simplified statement perhaps, since no mention was made of fluid dynamic concepts required for optimized flow and the still unformed studies for the ultimate designed prosthetic device.

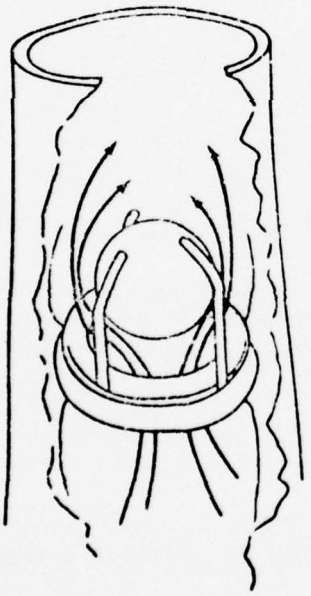


Fig. 5: Diagram Showing Flow Direction Through A Ball Type Artificial Heart Valve.

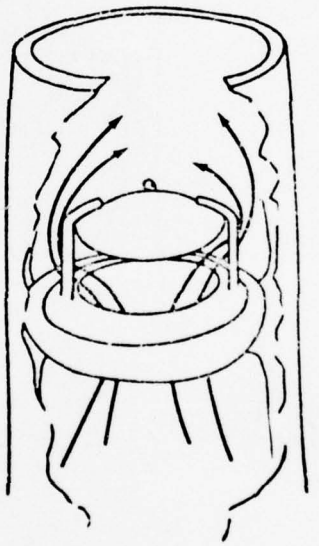


Fig. 6: Diagram Showing Flow Direction Through A Disc-Type Artificial Heart Valve.

This most widely used artificial valve, the Starr-Edwards type, has a silicon rubber (silastic) ball which is forced away from the valve ring position for desired flow and which is seated against the valve ring to prevent reverse flow at the proper times. The cage of this valve is bulky, however, causing a problem with mitral valve replacement for some patients because of their left ventricular chamber contour when in the collapsed position. The cage may cause septal endocardial irritation and cardiac arrhythmias while the ball could interfere in the left ventricular outflow. In turn, the valve, in its entirety, may interfere with myocardial contraction and be related to an ineffective cardiac output. Maximum hemodynamic benefit is probably nullified and thromboembolism may be realized due to valvular obstruction. Post-operative autopsies have shown linear myocardial rupture along the posterolateral strut of the valve cage with endocardial grooving present in the vicinity of other cage struts. Also, in mitral regurgitation patients the ventricular septum may be hypertrophied. Since this would cause shortening of the distance between the mitral annulus and the septum in the outflow tract region, the conduction system may contact the artificial valve's cage which is directed towards the septum, with trauma resulting. Too, in all valves, an interface traps red blood cells when in the closing phase. To prevent hemolysis, natural valve components allow broad surface contact between pliable elastic tissues. Hemolysis, however, is aided by a rigid ball-type valve diaphragm striking a metal seat. It is also possible for the ball to rotate within the valve cage, possibly damaging blood cells, although some investigators feel that large shearing stress, present between the ball and cage struts, allows blood damage to occur.

Ball variance in the Starr-Edwards valve has led to the replacement of the silicon rubber ball in some models by a hollow metal poppet, made of an alloy called Stellite*, noted for high strength and corrosion resistance. Also, the cage struts and the seat of the valve have now been covered by Teflon cloth in an attempt to form tissue ingrowth and prevent embolism formation. The long-term effect of tissue ingrowth is unknown and thromboembolism is still present. It is noted that some experimentation is being performed at a few installations with vitrous carbon or stainless steel for ball replacement.

As to refinements of the ball valve design, Magovern [8] designed a prosthesis with multiple pin fixation, as shown in figure 7.

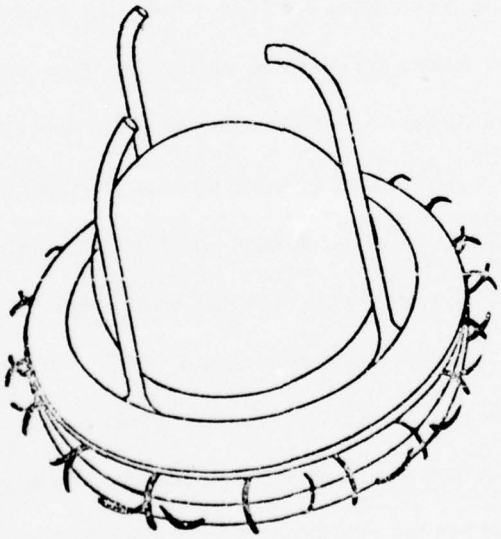


Fig. 7: Magovern Artificial Heart Valve

* Stellite: an alloy consisting of Cobalt (approx. 61%), Chromium (25.5-29%), Molybdenum (5-6%), Nickel (1.75-3.75%).

The pins replace sutures at the valve ring position since sutures are time-consuming, tedious, and prolong the duration of bypass of blood circulation through the heart-lung machine during surgery. Although success with this valve is comparable to the Starr-Edwards type, it is less widely used due to incidences of infection, detachment of pins, and bending of pins if calcified deposits are present at the valve seating position. Another refinement concept was sought by development of the Smeloff-Cutter Full Flow Orifice valve [9]. In this design, as seen in figure 8, a double-caged valve was designed to increase the orifice-to-ball diameter ratio so as to allow more of an orifice for flow.

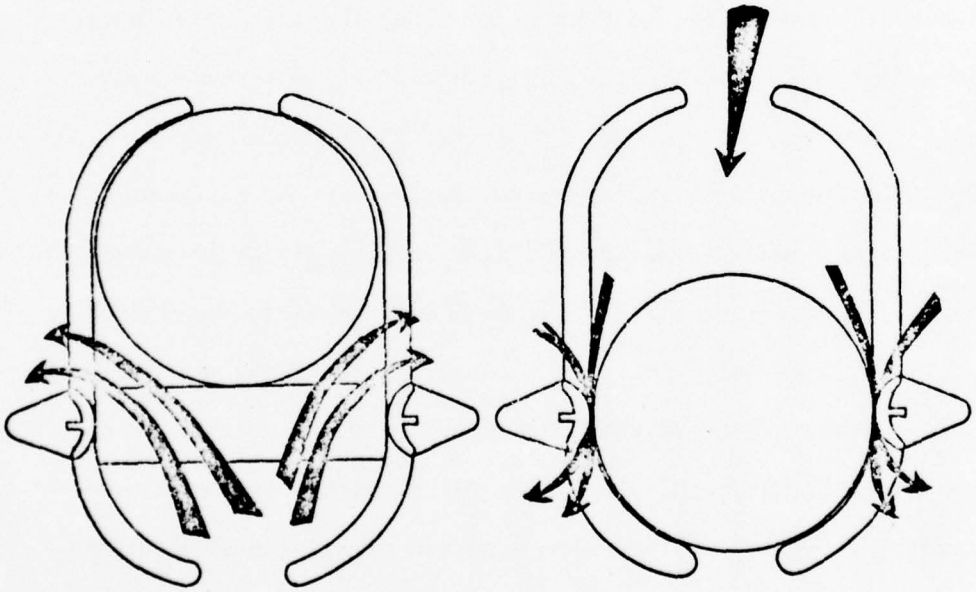


Fig. 8: Smeloff-Cutter Full Flow Orifice Artificial Heart Valve

Since the diameter of the ball is similar to the diameter of the orifice, obstruction to blood flow across such a valve was assumed to be minimal. The appearance of the second cage upstream of the orifice allows stoppage of motion of the ball rather than having the sphere seat on the orifice rim. Too, open-ended cages are intended to prevent areas of stagnation which can lead to embolism formation. In practice a disadvantage can occur with such a design that is not present in the Starr-Edwards configuration. If a small change occurs in the configuration of the ball, the ball can impact in the inflow orifice. Clinical trials have not been as successful as in vitro experiments.

With the advent of the ball-type prosthetic heart valve also came the realization that ball valve cages are bulky, causing a problem in some patients because of the contour of the left ventricular chamber, besides the aforementioned hypertrophy problem. A low profile valve is shown in figure 6. One variance, however, is an angled one such as the Hammersmith type previously shown in figure 3b. At the present time there are perhaps a score of such disc valves; all with low profile cages. Again, it was Hufnagel [10] that proposed the first valve of this nature in 1965. This valve included a polypropylene frame and silicone coated disc. Other types are the Cross-Jones valve [11] with an open-ended cage and silicone rubber lens, reinforced with a titanium inner ring (figure 9), the Kay-Shiley valve with two parallel struts and a plain silicone rubber disc (figure 10), the Melrose-Alvarez valve [13] with a polypropylene ring and occluder (figure 11), the Gott valve [14] with heparin-benzyl-coated silicone hinge flaps (figure 12) and the Davile valve [15] with felt ring and polypropylene occluder; figure 13.

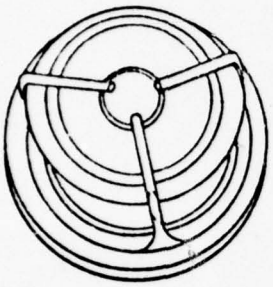


Fig. 9: Cross Jones Artificial Heart Valve

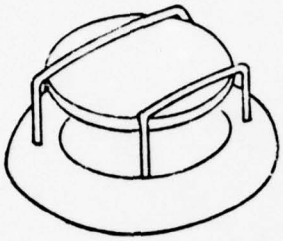


Fig. 10: Kay-Shiley Artificial Heart Valve

Fig. 11: Melrose-Alvarez Artificial Heart Valve

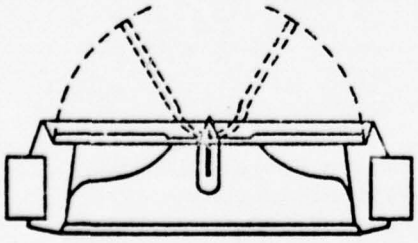


Fig. 12: Gott Artificial Heart Valve

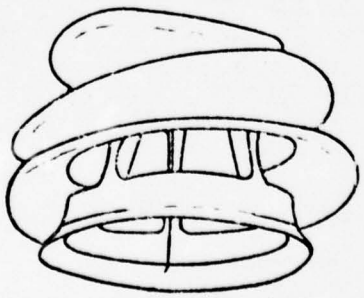


Fig. 13: Davila Artificial Heart Valve

Slight variations in the shaping of the cage or the disc have allowed investigators to attach their names to various models of the discoid-type valve. Without further description, a few more examples are shown below in order to allow the reader to form his opinion of the position that the surgeon finds himself in if he truly attempts to evaluate the presently available valves.

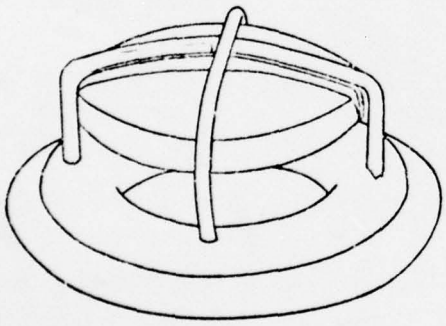


Fig. 14: Starr-Edwards Disc-Type Artificial Heart Valve

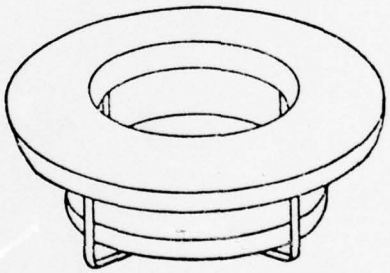


Fig. 15: Harken Mitral Discoid-Type Artificial Heart Valve

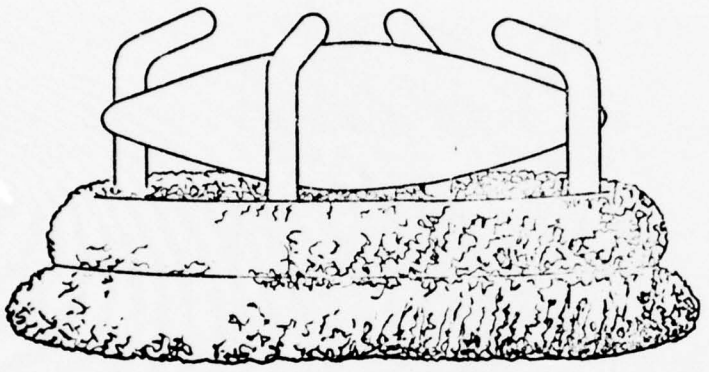


Fig. 16: Cooley-Bloodwell-Cutter Artificial Heart Valve

To compound the problem of proper choice, there are also design variations such as the hinged leaflet shown in figure 17. Two such examples of this style are the Lillehei-Kaster Pivoting disc [16], figure 18, and the Wada-Cutter valve [17], figure 19. Personal experience has shown flow defects in the Wada valve.

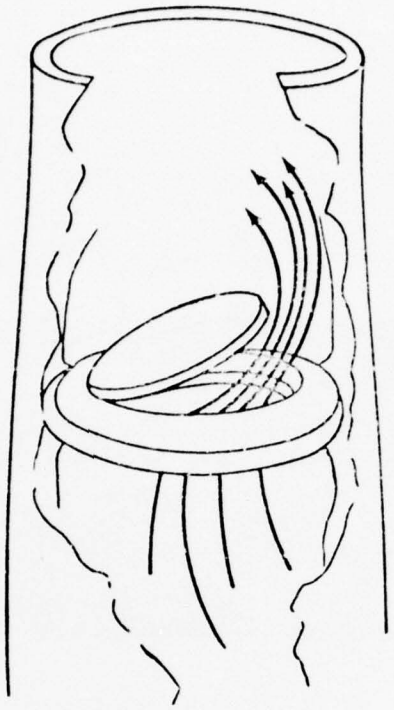


Fig. 17: Diagram Showing Flow Direction Through A Hinged-Leaflet Type Artificial Heart Valve

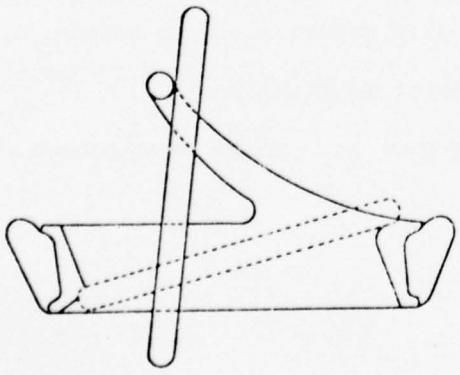


Fig. 18: Lillehei-Kaster Pivoting Disc-Type Artificial Heart Valve

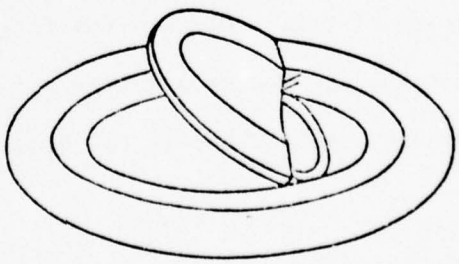


Fig. 19: Wada-Cutter Artificial Heart Valve

Some study has also been accomplished employing a true poppet type valve as shown in figures 20 and 21. The Barnard UCT A06 valve is of this type. Vorhauer and McElhaney [18] developed a tear-drop shaped artificial heart valve which has some merit due to its streamlining for aid to blood flow, however, until the original design is amplified, the design performs a disfavor to a collapsed ventricle by the length and sharpness of the occluder and protective vanes. In fairness, it should be mentioned that the developers propose it only as an aortic valve replacement.

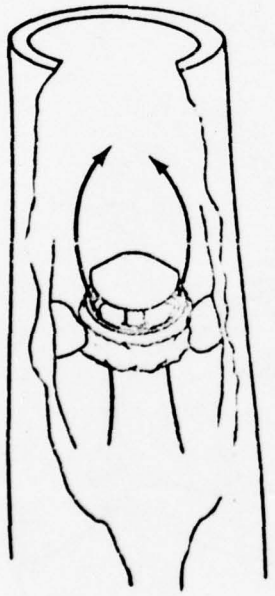


Fig. 20: Diagram Showing Flow Direction Through A Poppet-Type Artificial Heart Valve

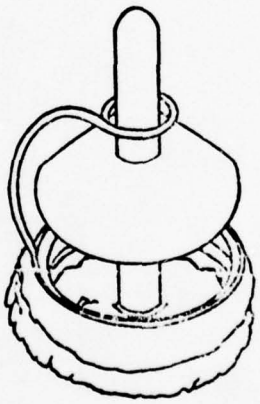


Fig. 21: Barnard UCT Artificial Heart Valve

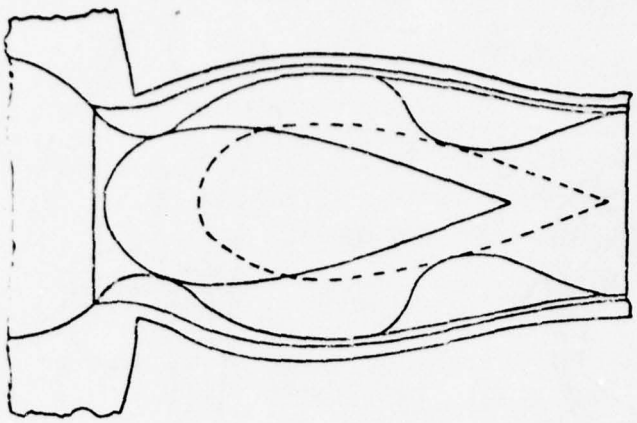


Fig. 22: Vorhauer Artificial Heart Valve

A final example of the many design variations of the prosthetic heart valve is the Nakib toroidal heart valve [19] as shown in figure 23. This is an interesting design since it includes a titanium ring that is pushed away from the centered and stationary poppet upon opening. This allows central and also peripheral flow, yet good sealing for opposite direction blood flow. There is also a reduction in occluder weight in comparison to a comparable ball-type valve. Too, the poppet is somewhat streamlined for proper flow effects. It has an open-end low profile cage, however, little follow-up of patient implantation is available and no analytic studies have been seen by this writer.

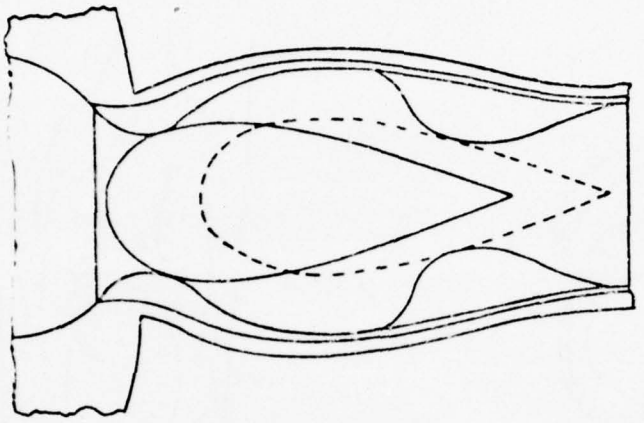


Fig. 22: Vorhauer Artificial Heart Valve

A final example of the many design variations of the prosthetic heart valve is the Nakib toroidal heart valve [19] as shown in figure 23. This is an interesting design since it includes a titanium ring that is pushed away from the centered and stationary poppet upon opening. This allows central and also peripheral flow, yet good sealing for opposite direction blood flow. There is also a reduction in occluder weight in comparison to a comparable ball-type valve. Too, the poppet is somewhat streamlined for proper flow effects. It has an open-end low profile cage, however, little follow-up of patient implantation is available and no analytic studies have been seen by this writer.

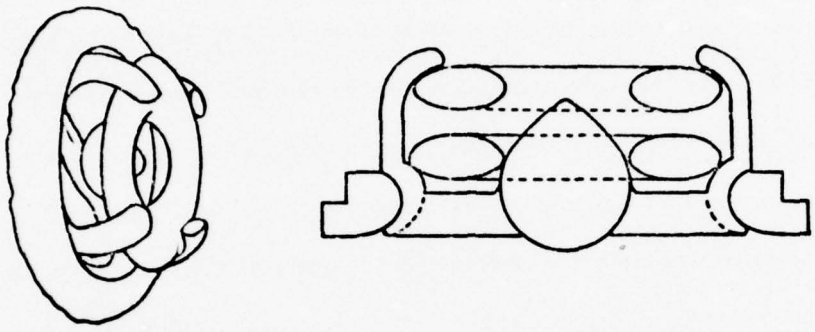


Fig. 23: Nakib Toroidal Artificial Heart Valve

Finally, one notes that the arguments put forth for either the ball or disc type designs appear strong for either side. It has been suggested that lens (disc) wear is not as evenly distributed as ball wear under comparable stress situations [20]. Since a ball tends to spin during each cycle, the full impact of the stress is both reduced and distributed more evenly. In addition, a disc tends to move up and down more often than it rotates, and also strikes the frame harder during each cycle. At present, these are theoretical objections which are not verified by clinical tests as yet. Interestingly, uneven wear of the disc has not been shown clinically when this valve type is in the mitral position, however, degeneration of the disc, when made of a silicon, is seen when

it is in the aortic position. On the other hand, some studies have shown a functional superiority of the disc valve over the ball valve, especially in the mitral section. The disc valve opens and closes more quickly, there is less regurgitation during the closing phase and the lens strikes the valve seat with less force, which may negate a previous theoretical argument. It also has a shorter traveling distance than the ball valve.

One notes that in the ball valve, chemical analysis has shown that for the implantation time of some duration, a few valves have undergone biochemical change which can then result in mechanical change. Such a variant ball has revealed the presence of free fatty acids, glyceryl fatty acid esters, methyl esters of fatty acids, cholesterol fatty acid esters, and palmitic acid. Also, rosette type crystals, not a by-product of vulcanization, have been found. It has also been shown by the leading manufacturer of this type of heart valve that an affected ball shows cracks or fissures well below the surface of the ball. It is possible then that the biological change occurs with subsequent deposit of lipid material and crystals, and a large resulting deep fissure forms; the latter then spreads to the surface of the ball. A requirement appears to be then to study the stresses as formed by the action of the artificial heart valve components with a view towards minimizing such stresses and the subsequent results when designing the optimum valve. Surprinsingly perhaps, and certainly worrisome is the fact that lethal degeneration of the silastic ball in a Starr-Edwards prosthetic valve type is now coming to light in some patients after an implantation time of six and one half years [21]. Figure 24 shows deformed and fractures poppets which either remain immobile within the cage or split into fragments. In passing it is noted that the senior writer of this paper has incorporated experimental photostress studies in conjunction with the computer graphics

studies so as to gather further of concern to stress problems, amongst other requirements.

In reiteration, considerable confusion exists in the minds of surgeons as to which heart valve substitute will eventually prove most preferable.

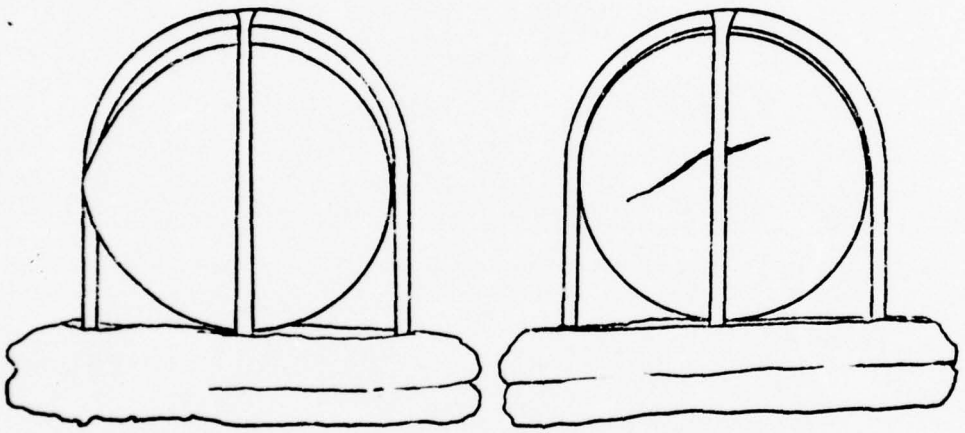


Fig. 24: Lethal Mitral Valve Variance

PRECEDING PAGE BLANK-NOT FILMED

CHAPTER IV

PROBLEM AREA II: ATHEROSCLEROSIS

Arteriosclerosis is the term given to a chronic disease characterized by abnormal thickening and hardening of the arterial walls. Our interest, however, is in atherosclerosis which is a particular form of arteriosclerosis formed by the deposition of fatty substances in and fibrosis of the inner layers of the arteries. Since the presence of atherosclerotic conditions is one of the prime causes of heart disorder and circulatory problems, then this fact, plus the knowledge that 500,000 people in the United States alone die of heart attacks (plus 200,000 of strokes), convinces one that intensified studies of this disease are indeed important. Coupling this conviction with the thought that a number of investigators [22,23] suggest a relationship between hemodynamic factors and atherosclerosis development allows one to form a fluid dynamic-computer graphic investigation.

It is startling to realize that laboratory studies with experimental animals and postmortem studies of human infants have shown that development of atherosclerosis often appears shortly after birth [24]. Extensive formations of atherosclerotic plaques were found in American casualties of the Korean war, as young as a median age of 23 years. At the other end of a history spectrum, typical atherosclerotic lesions have been found in the aortas of 3500 year old Egyptian mummies.

Campbell Moses [25] has reviewed certain historical investigations that lend agreement to the thesis that mechanical and physical factors may play roles as primary causative factors in the development of atherosclerosis. As examples, Carrel and Guthrie [26] in 1906 attempted to relate arterial pressure to acceleration of atherosclerosis, while Klotz [27] in

1908 studied uncomplicated hypertension and described hyperplasia of both the intima and muscular elements of the studied media. In 1914, Anitschkow [28] discussed the accelerating role of hypertension concerning experimental atherosclerosis, in the presence of hypercholesterolemia. Allbutt in 1915 [29] went further by claiming that hypertension is a prime factor in atherosclerosis. Beginning to somewhat employ mechanics, Duguid [30] in 1926 discussed lesion localization at the entrance to branch arteries when shearing stress occurred in the main vascular channel. More recently, Willis [31] investigated localizing factors in atherosclerosis with conclusions that hydrostatic pressure and/or lumen radius and curvature are favorable factors for local atherosclerotic formations.

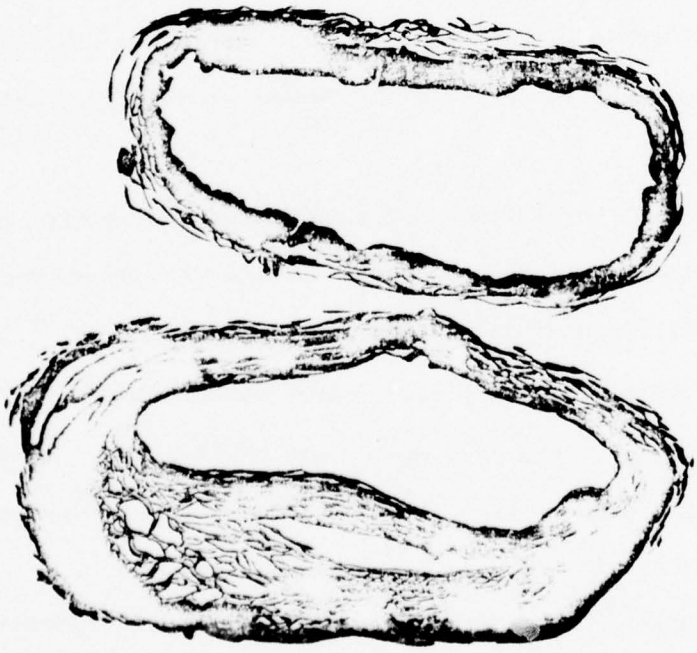


Fig. 25: Diseased Atherosclerotic Human Artery (Below) As Compared To A Healthy One (Above)

Of closer interest to our own approach are the continuing studies of Texon [32-35]. His hemodynamic investigations inevitably point toward the laws of fluid mechanics as the prime "raisons d'être" for localization of atherosclerotic lesions at specific areas of predilection in the arterial tree as well as for their absence in identical adjacent areas of intima. Such areas of predilection are specifically declared to be sites of diminished lateral or static pressure which occur in arteries at sites of tapering, bifurcation, branching, curvature and external attachment. Combinations and geometric variances of such areas add to the sum of such patterns. With such a localized decrease in static pressure at a site, occurence of a suction effect upon the wall is seen, with subsequent subjection of the arterial intima to the lifting effect of blood flowing past upon the endothelial layer and subjacent cells. Initially, the response is a reactive thickening by proliferation of endothelial cells and fibroblasts from subjacent layers. Successively, continuing blood flow enlarges such an initial plaque, lateral pressure has continual dimunition, and cellular elements and lipids are added to successive layers of fibroblastic proliferation. When such pathologic degeneration reaches a critical point, portions of the atherosclerotic plaques may detach from the arterial wall, allowing embolization and thrombosis. With such detachment the moving portion of a plaque may travel until it reaches a section of a smaller artery and plugs it. If this plugging occurs in the coronary artery it produces a heart attack. Occurrence in the brain causes a cerebral stroke while gangrene is the result of blockage in an artery of the lower extremities.

In the above short expose there has not been the attempt to assume that the bases of sites for predilection for atherosclerotic plaques are completely explained by fluid mechanics concepts. There are many medical

investigators who accept as conclusive the reported evidence of a relationship between atherosclerosis and elevated serum cholesterol levels. The possible consequences of diet and hereditary factors concurrent with elevated serum cholesterol levels are duly noted, however, our interest and capabilities lie with fluid mechanics arguments for atherosclerotic pathogenesis.

CHAPTER V

MATHEMATICAL APPROACH

A. Some Preliminary Fluid Flow Concepts

In initiating hemodynamic studies of the circulation a useful preliminary concept is to assume a long straight tube with a steady flow of fluid inside. Maintaining this steady flow is a constant pressure which is required because of the fluid's inherent viscosity. Poiseuille, a physician, in 1842, was the author of an experimental law governing the flow of such an encased fluid. Poiseuille stated that the head of the required pressure is directly proportional to pipe length, flow rate and viscosity, and inversely proportional to the fourth power of the radius. (Girard in 1813 had assumed a third power law which was subsequently corrected by Hagen in 1839.) Upon the injection of dye into the liquid for a visual aid, it was noted that along the pipe axis, liquid motion was quicker than at the walls. The explanation was that the fluid particles moved in a series of parallel laminae; the lamina of fluid closest to the pipe wall was stationary and each successive outward lamina slipped against the viscous friction of its neighbor, which had progressively lessened as one approached the pipe axis. Such a group of parallel laminaes of fluid constitutes laminar flow or streamlines since particles of flow in one such ideal streamline do not become part of another parallel streamline of flow. It is of historical interest to note that Poiseuille, as a physician, was interested particularly in blood circulation. By choosing glass tubes of capillary size for his flow experiments he was fortunate in obtaining laminar flow to an accurate degree which would not have necessarily occurred in arterial size tubes. He also chose water as the working fluid since he could not prevent blood from coagulating, and again was fortunate for he

thus avoided the anomalous viscous properties present in blood. It is now known, however, that if the flow rate is increased, there is a point reached where the neat pressure-flow relationship breaks down and laminar flow ceases. Fluid particles then follow random paths across the tube as well as a main movement through the tube, with the result that flow becomes turbulent.

The rigid pipe concept is a preliminary one for arterial flow studies. The Poiseuille law that one usually encounters has been placed in a form that is amenable to Navier's theoretical equations for viscous flow, which were later amplified by Stokes in the 1840's. Poiseuille's equation is questionable when it is applied to the vascular system. In the vicinity of branch points the velocity profile may not fit the one formed by the equation. In arteries, the flow may not be irrotational, while in the veins, blood may appear as a non-Newtonian fluid, wherein apparent viscosity decreases with a linear velocity increase and a vessel diameter increase. For higher hematocrit values there are greater viscosity changes.

B. Determination of Flow About an Obstacle.

Turbulence is one of the most important aspects of the science of fluid mechanics and fluid dynamics, and one of the least understood; a phenomenon that defies attempts at complete analysis. Rarely is it possible to produce fluid flow having considerable gross movement but no turbulence. Although laminar flow may be desirable and achievable, the presence of a fine scale of fluctuations, still having turbulent properties, may be inherent. To compound the problem, the mathematical treatment of fluid motion has led to a set of simultaneous nonlinear partial differential equations of the second order; the Navier-Stokes equations, together with the equation of continuity. These Navier-Stokes equations of motion are

presently intractable in that no mathematical technique is available for treating the difficulty of nonlinearity. Known particular solutions are those exact ones related to certain limiting cases of very low speeds and the inviscid fluid situation with irrotational motion. However, modified Navier-Stokes equations may be solved by various methods employing boundary layer theory, in certain instances.

Wind tunnel experiments show that a fluid behaves in a series of fairly regular changes when flowing around a fixed object in a stream of motion. At low velocity, laminar flow exists about the object up to a point of critical speed where the onslaught of turbulence is then noticed. Vortices are shed alternately from the rear top and bottom sections of the immersed object, resulting in an oscillating wake familiarly known as a Von Karman vortex street. Further increase in flow velocity causes the vortices to deteriorate into random eddies, signifying fully developed turbulence. Thus, if such a pattern can be developed by use of a computer and subsequently plotted or displayed optically, a calculated experiment may be formed.

1. Navier-Stokes Equations Forms Required.

The forms of the Navier-Stokes equations required for the present investigation are those that describe two dimensional, incompressible viscous fluid flow, namely;

$$\rho\left(\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right) = -\frac{\partial p}{\partial x} + \mu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \qquad (1)$$

$$\rho\left(\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}\right) = -\frac{\partial p}{\partial y} + \mu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \qquad (2)$$

where ν is the kinematic viscosity, and where the x and y symbols are the usual Cartesian coordinates. Pressure is p and t is time. The velocity components along the x and y axes are u and v.

The equation of continuity is

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} = 0 \tag{3}$$

which is equivalent to

$$\frac{d\rho}{dt} + \rho\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) = 0 \tag{4}$$

where

$$\frac{d\rho}{dt} = \left(\frac{\partial}{\partial t} + u\,\frac{\partial}{\partial x} + v\,\frac{\partial}{\partial y}\right)\rho \tag{5}$$

These equations express the conservation of mass. Assuming the fluid to be incompressible as noted, then $\frac{d\rho}{dt} = 0$, and the equation of continuity reduces to

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{6}$$

The stream function ψ is defined in terms of the velocity components so that it satisfies the equation of continuity. Thus

$$u = \frac{\partial \psi}{\partial y} \tag{7}$$

and

$$v = -\frac{\partial \psi}{\partial x} \tag{8}$$

The vorticity ω is defined as

$$\omega = -\frac{1}{2}\left(\frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}\right) \tag{9}$$

If one differentiates equation (1) with respect to y and equation (2) with respect to x, and notes that ρ is constant, then substracts, the vorticity transport equation is formed;

$$\frac{d\omega}{dt} = \frac{\partial \omega}{\partial t} + \frac{\partial \omega}{\partial x}\frac{\partial \psi}{\partial y} - \frac{\partial \omega}{\partial y}\frac{\partial \psi}{\partial x} = \nu \nabla^2 \omega \tag{10}$$

The stream function is useful in determining flow streamlines and the vorticity function is a function of fluid velocity in the vertical and horizontal axes directions. As seen by equations (7) and (8), the vorticity function is usually transformed into stream function values to ease the solving of the fluid motion equations. With the aid of the stream and vorticity functions the newly formed vorticity transport equation allows one to have a fluid diffusion equation in terms of vorticity.

An equation for p may be obtained by differentiating equation (1) with respect to x and equation (2) with respect to y. The equation formed is

$$\nabla^2 p = 2\rho \left[\frac{\partial^2 \psi}{\partial x^2} \frac{\partial^2 \psi}{\partial y^2} - \frac{\partial^2 \psi}{\partial x \partial y} \right] \tag{11}$$

On a fixed solid boundary

$$u = v = 0, \tag{12}$$

however, in terms of the stream function

$$\frac{\partial \psi}{\partial s} = \frac{\partial \psi}{\partial n} = 0 \tag{13}$$

where s and n refer to the tangential and normal directions respectively. Employing equations (1), (2), (6) and (9), one sees that on the boundary of a solid

$$\frac{\partial p}{\partial n} = 2\mu \frac{\partial \omega}{\partial s} \qquad \frac{\partial p}{\partial s} = -2\mu \frac{\partial \omega}{\partial n} \tag{14), (15}$$

If it is now assumed that U is a constant velocity and L is a characteristic length of the system, then dimensionless variables, denoted with a subscript d, are introduced as follows:

$$\psi_d = \frac{\psi}{\nu}$$

$$x_d = \frac{x}{L}$$

$$t_d = -\frac{\nu}{L^2} t$$

$$u_d = \frac{L}{\nu} U$$

$$\omega_d = \frac{L^2}{\nu} \omega \tag{16}$$

where ν is the kinematic viscosity as before. The vorticity transport equation may now be written as

$$\frac{\partial \omega}{\partial t_d} + \frac{\partial \psi}{\partial y_d}\frac{\partial \omega}{\partial x_d} - \frac{\partial \psi}{\partial x_d}\frac{\partial \omega}{\partial y_d} = \nabla^2 \omega_d \tag{17}$$

and the vorticity equation becomes

$$\nabla^2 \psi_d = -2\omega_d \tag{18}$$

The subletters attached to the dimensionless variables will be dropped for convenience and all equations will then be assumed to be expressed in terms of dimensionless variables.

The boundaries involve the specification of the velocity of the fluid at all boundaries. To see how the boundary conditions are specified, the simple example of fluid flowing over the surface of a flat plate is noticed at this point, however, boundary conditions will be more fully discussed later. Thus,

$$\begin{aligned} u_p(x,0,t) &= 0 \\ v_p(x,0,t) &= V_p\, g(t)\, f(x) \end{aligned} \tag{19}$$

(where p stands for physical variables) are imposed on a boundary at $y = 0$, then the dimensionless problem should have the boundary conditions

$$\frac{\partial \psi}{\partial y}(x,0,t) = 0 \tag{20}$$

$$\psi(x,0,t) = \text{const} + \frac{UL}{\nu}\, g(t) \int_0^x f(\alpha)\, d\alpha \tag{21}$$

where α is a dummy variable. The boundary conditions contain the dimensionless term UL/ν. This is usually seen as R_e which is the Reynolds number. The Reynolds number enters the problem through the boundary conditions. The dimensionless function $g(t)$ allows a velocity change with time on the boundary. It is usually seen in the form

$$g(t) = (1-\varepsilon^{-at})$$

The dimensionless function $f(\alpha)$ allows spacial variation of the velocity on the boundary.

The initial conditions are

$$u(x,y,0) = 0$$

$$v(x,y,0) = 0. \qquad (22)$$

Thus, in equations (17) and (18) one has two simultaneous second order partial differential equations in two unknowns ψ and ω with boundary and initial conditions of the forms (20, (21), and (22). Equation (17) is a nonlinear equation which allows analytical solutions for only very simple flow problems.

2. Computational Mesh

To perform the desired flow calculations an obstacle was placed in the center of the flow stream. The area about the obstacle was divided into square cells, which, in their entirety, formed a basic computing mesh. An optimum arrangement of cells was formed. The fixed object, about which flow was to be studied, was placed with its boundaries along mesh lines. Enough cell spacing was allowed upstream of the obstacle for initiation of flow perturbations and initial conditions as required. (See figure 26)

Figure 27 shows an enlarged portion of the computational mesh.

All partial differential equations involved had their derivatives replaced by finite difference equivalents [36], which allowed a series of finite difference equations instead of the difficult differential equations. These finite difference approximations are algebraic in form and relate the value of the dependent variables at a point in the mesh to the values at a number of symmetrically arranged neighboring mesh points, at one instant of time and then succeeding time intervals. Although the method sounds time consuming, and can be, if performed by hand for approximately the 1500 cells available in the present mesh network, the method becomes

quite routine and immensely accelerated when the method of squares [37] or the relaxation method [38] is applied with their ramifications, and in turn, programmed for the digital computer. Calculation times are then matters of seconds. Values of the velocity components were calculated as centered differences of the stream function at each cell corner.

Figure 28 shows the irregular computing mesh employed so as to allow generalized dimensional quantities for forming the difference equations. As will be seen, this generalized concept was useful for developing the theory required to analyze obstacles that consisted of curved boundaries.

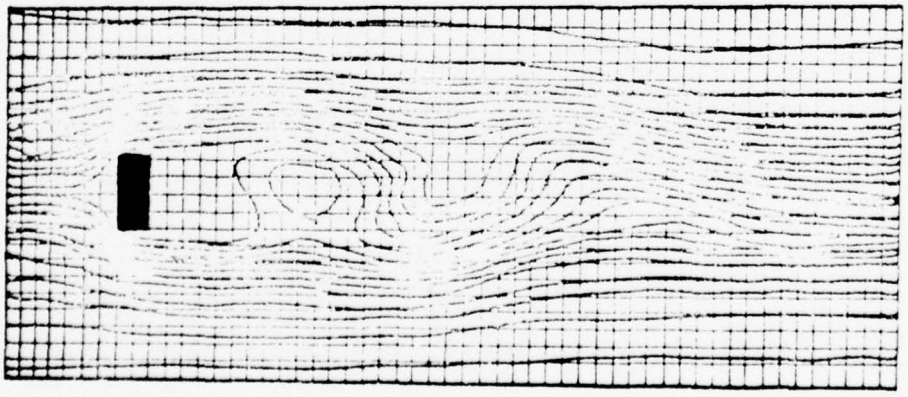


Fig. 26: An Idealized Obstacle and its Computational Mesh

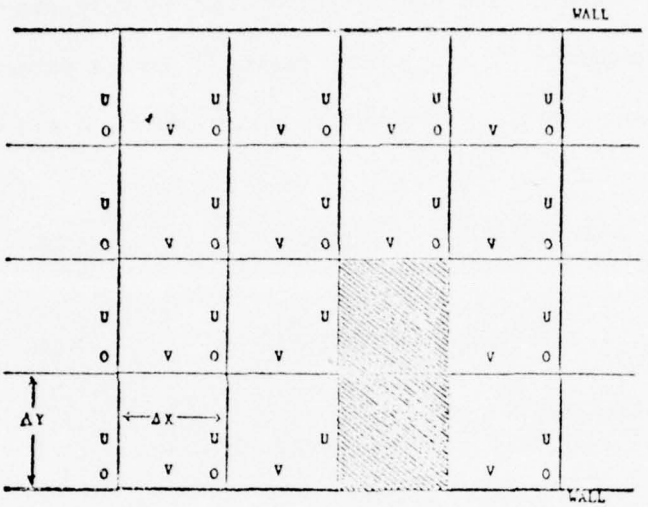


Fig. 27: Portion of Computational Mesh
o = stream function.
u,v = velocity components.

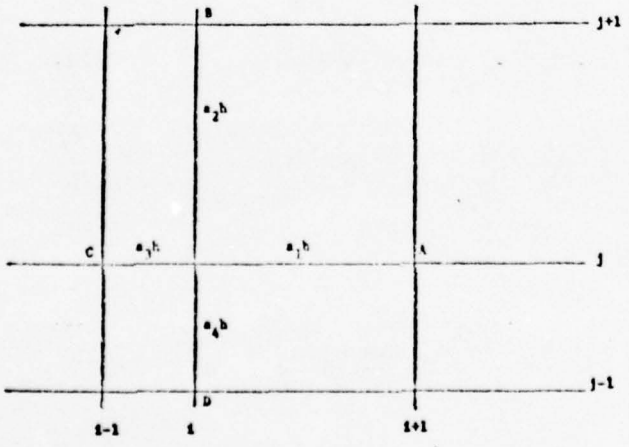


Fig. 28: The Irregular Computational Mesh

3. Finite Difference Equations Required.

It was first necessary to form finite difference approximations to the vorticity transport equation and the equation linking vorticity and the stream function. Expanding by means of a Taylor's series formation and referring to figure 28, required evaluation about points A and C are seen to be

$$u_A = u_O + a_1 h u_{O_x} + 1/2 a_1^2 h^2 u_{O_{xx}} + \mathbf{O}(h^3) \tag{23}$$

$$u_C = u_O - a_3 h u_{O_x} + 1/2 a_3^2 h^2 u_{O_{xx}} + \mathbf{O}(h^3) \tag{24}$$

where u is a dummy variable for ψ or ω.

Eliminating $u_{O_{xx}}$ one obtains

$$u_{O_x} = (a_3^2 u_A + (a_1^2 - a_3^2) u_O - a_1^2 u_C)/h a_1 a_3 (a_1 + a_3) \tag{25}$$

by ignoring the higher orders of h.

Elimination of u_{O_x} allows

$$u_{O_{xx}} = 2(a_3 u_A - (a_3 + a_1) u_O + a_1 u_C)/h^2 a_1 a_3 (a_1 + a_3) \tag{26}$$

In a similar manner,

$$u_B = u_O + a_2 h u_{O_y} + 1/2 a_2^2 h^2 u_{O_{yy}} + \mathbf{O}(h^3) \tag{27}$$

$$u_D = u_O - a_4 h u_{O_y} + 1/2 a_4^2 h^2 u_{O_{yy}} + \mathbf{O}(h^3) \tag{28}$$

Eliminating $u_{O_{yy}}$ one obtains

$$u_{O_y} = (a_4^2 u_B + (a_2^2 - a_4^2) u_O - a_2^2 u_D)/h a_2 a_4 (a_2 + a_4) \tag{29}$$

and similarly

$$u_{O_{yy}} = 2 (a_4 u_B - (a_2 + a_4) u_O + a_2 u_D)/h^2 a_2 a_4 (a_2 + a_4) \tag{30}$$

Equations (23) through (30) are then used to obtain the finite difference approximations to the nondimensional forms of equations (17) and (18). Placing equations (25), (26), (29), and (30) in slightly different forms

$$\left.\frac{\partial u}{\partial x}\right|_O = \frac{a_3^2 u_A + (a_1^2 - a_3^2) u_O - a_1^2 u_C}{h a_1 a_3 (a_1 + a_3)} \tag{31}$$

$$\left.\frac{\partial^2 u}{\partial x^2}\right|_O = \frac{a_3 u_A - (a_3 + a_1) u_O + a_1 u_C}{\frac{h^2}{2} a_1 a_3 (a_1 + a_3)} \tag{32}$$

$$\left.\frac{\partial u}{\partial y}\right|_O = \frac{a_4^2 u_B + (a_2^2 - a_4^2) u_O - a_2^2 u_D}{h a_2 a_4 (a_2 + a_4)} \tag{33}$$

$$\left.\frac{\partial^2 u}{\partial y^2}\right|_O = \frac{a_4 u_B - (a_2 + a_4) u_O + a_2 u_D}{\frac{h^2}{2} a_2 a_4 (a_2 + a_4)} \tag{34}$$

Recalling that the vorticity and stream function relationship is

$$\nabla^2 \psi = -2\omega = \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2}$$

and the vorticity transport equation is

$$\frac{\partial \omega}{\partial t} = \nabla^2 \omega + \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y} - \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x}$$

or

$$\frac{\partial \omega}{\partial t} = \frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} + \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y} - \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x}$$

and that the nodes A,B,C,D, and O of each mesh cell have the following positions in the i.j notation

A = i+1,j B = i,j+1 C = i-1,j D = i,j-1 O = i,j

then by substitution, (with the dummy variable u becoming ψ and ω in the proper places, the vorticity transport equation in finite difference form is;

$$\frac{\partial\omega}{\partial t} - \left[\frac{a_3\omega_{i+1,j} - (a_3+a_1)\omega_{i,j} + a_1\omega_{i-1,j}}{\frac{h^2}{2}\ a_1a_3(a_1+a_3)}\right]$$

$$- \left[\frac{a_4\omega_{i,j+1} - (a_2+a_4)\omega_{i,j} + a_2\omega_{i,j-1}}{\frac{h^2}{2}\ a_2a_4(a_2+a_4)}\right]$$

$$- \left[\frac{a_3^2\psi_{i+1,j} + (a_1^2-a_3^2)\psi_{i,j} - a_1^2\psi_{i-1,j}}{ha_1a_3(a_1+a_3)}\right]$$

$$\times \left[\frac{a_4^2\omega_{i,j+1} + (a_2^2-a_4^2)\omega_{i,j} - a_2^2\omega_{i,j-1}}{ha_2a_4(a_2+a_4)}\right]$$

$$+ \left[\frac{a_3^2\omega_{i+1,j} + (a_1^2-a_3^2)\omega_{i,j} - a_1^2\omega_{i-1,j}}{ha_2a_4(a_2+a_4)}\right]$$

$$\times \left[\frac{a_4^2\psi_{i,j+1} + (a_2^2-a_4^2)\psi_{i,j} - a_2^2\psi_{i,j-1}}{ha_1a_3(a_1+a_3)}\right] \qquad (35)$$

Regrouping:

$$\frac{\partial\omega}{\partial t} - \left\{\frac{2}{h^2a_1(a_1+a_3)}\ \omega_{i+1,j} - \frac{2}{h^2a_1a_3}\ \omega_{i,j} + \frac{2}{h^2a_3(a_1+a^3)}\ \omega_{i-1,j}\right\}$$

$$- \left\{\frac{2}{h^2a_2(a_2+a^4)}\ \omega_{i,j+1} - \frac{2}{h^2a_2a_4}\ \omega_{i,j} + \frac{2}{h^2a_4(a_2+a_4)}\ \omega_{i,j-1}\right\}$$

$$- \left\{\frac{a_3h}{h^2a_1(a_1+a_3)}\ \psi_{i+1,j} + \frac{(a_1-a_3)h}{h^2a_1a_3}\ \psi_{i,j} - \frac{a_1h}{h^2a_3(a_1+a_3)}\ \psi_{i-1,j}\right\}$$

$$\left\{\frac{a_4 h}{h^2 a_2(a_2+a_4)}\,\omega_{i,j+1} + \frac{(a_2-a_4)h}{h^2 a_2 a_4}\,\omega_{i,j} - \frac{a_2 h}{h^2 a_4(a_2+a_4)}\,\omega_{i,j+1}\right\}$$

$$+\left\{\frac{a_3 h}{h^2 a_1(a_1+a_3)}\,\omega_{i+1,j} + \frac{(a_1-a_3)h}{h^2 a_1 a_3}\,\omega_{i,j} - \frac{a_1 h}{h^2 a_3(a_1+a_3)}\,\omega_{i-1,j}\right\}$$

$$\left\{\frac{a_4 h}{h^2 a_2(a_2+a_4)}\,\psi_{i,j+1} + \frac{(a_2-a_4)h}{h^2 a_2 a_4}\,\psi_{i,j} - \frac{a_2 h}{h^2 a_4(a_2+a_4)}\,\psi_{i,j+1}\right\}$$

$$= 0 \qquad (36)$$

In turn the stream function $\nabla^2\psi = -2\omega$ has the finite difference form

$$\frac{a_3\psi_{i+1,j} - (a_3+a_1)\psi_{i,j} + a_1\psi_{i-1,j}}{\frac{h^2}{2}\,a_1 a_3(a_1+a_3)}$$

$$+\frac{a_4\psi_{i,j+1} - (a_2+a_4)\,\psi_{i,j} + a_2\psi_{i,j-1}}{\frac{h^2}{2}\,a_2 a_4(a_2+a_4)} = -2\omega_{i,j} \qquad (37)$$

Regrouping:

$$\frac{1}{a_1(a_1+a_3)}\,\psi_{i+1,j} - \frac{1}{a_1 a_3}\,\psi_{i,j} + \frac{1}{a_3(a_1+a_3)}\,\psi_{i-1,j}$$

$$+\frac{1}{a_2(a_2+a_4)}\,\psi_{i,j+1} - \frac{1}{a_2 a_4}\,\psi_{i,j} + \frac{1}{a_4(a_2+a_4)}\,\psi_{i,j-1} = -h^2\omega_{i,j} \qquad (38)$$

After the computer program calculated the type of each grid point, the program then calculated the mesh distances a_1, a_2, a_3 and a_4 for each node. As Booy pointed out [39], the analysis of a difference equation on points next to a boundary involves a finite difference stencil with arms (a_i, i = 1,2,3,4) of various lengths such as occurred in our problem. It was his inherent stipulation that no mesh distance a_i be less than 0.25h. It was therefore decided that for our program, if a mesh node was within 0.25h of the boundary, then that node was redesignated as a boundary point as shown in figure 29.

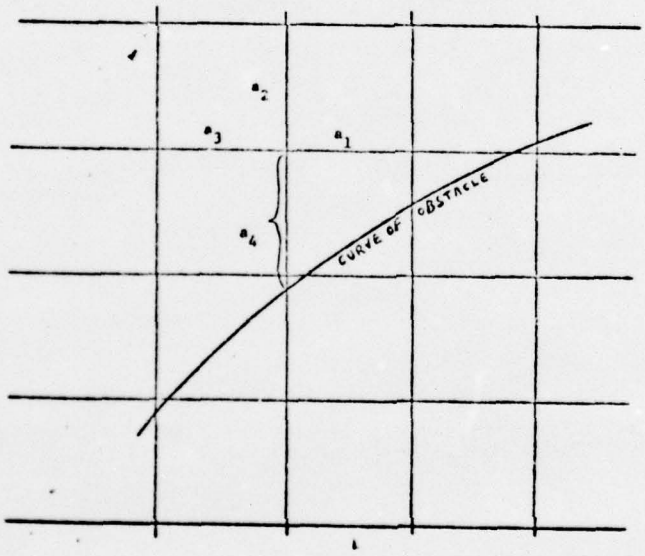


Fig. 29: Finite Difference Stencil Near A Curved Boundary.

4. Numerical Procedure

The details of the numerical procedure can be confusing if not followed closely, therefore a generalized view will first be given, followed by a detailed discussion which then culminates in the presentation of the first set of pictorial results.

At the start of a computational experiment values of the horizontal and vertical components of the average fluid velocity, together with vorticity and stream function values at each cell nodal point in space and at initial time must be specified. Then, ω, the vorticity function and ψ, the stream function are calculated at each corner of each cell with the knowledge of initial conditions of the flow. The vorticity transport equation, in its equivalent algebraic form, is then solved so as to replace initial vorticity values at all lattice points with advanced values in successive times. Such are used as source terms in the Poisson equation solved for the stream function. The equation for the stream function on the interior of the flow field is then solved as a boundary-value problem, with the boundary stream function specified by programmed instructions to the computer. Vorticity values are calculated in an iterative fashion on the boundaries of the obstacle formed within the flow field and the process is repeated until the maximum changes in vorticity and stream function at any nodal point in the field reach a prescribed tolerance of error within the computational process. After this tolerance has been reached the computer-programmed process is commenced again with new vorticity and stream function values on the boundary. One is thus allowed to visualize the flow pattern and vorticity surface about the obstacle for changing increments of time. The computer must, of necessity, be supplied with operating instructions for calculating changes

in flow configuration at any sequential instant, using the values previously mentioned as data input at each point. Numerical stability must be assured and the requirement of truncation turbulence not to be allowed, enforced. The operating instructions are necessarily tedious and time consuming since step-by-step instructions are required, yet the program is written with sufficient generality to solve a wide variety of similar problems. For example, for the heart valve study, an object in the stream flow may be studied with an approximately equivalent shape to a heart valve section of interest, but with the allowance of a family of like cardiac-valve shapes to be written into the computer program. Once the overall format is seen, operating instructions in terms of specific requirements can be initiated for a particularly shaped valve segment, but with the allowance for changing requirements and/or shaped segments. The program will allow for additional and immediate required evaluations at prescribed points in space or time. Ultimately, the results are presented in any amount of detail desired by means of an optical display, data print-outs, or single time-frame plots.

Detail-wise, the numerical procedure is a successor to the technique formed by Fromm [40] and Harlow [41]. Their procedure was basically as follows:

i) At the beginning of a cycle all required variable values were placed into computer memory slots as initial data or results from a previous cycle.

ii) Since vorticity and stream function values were assumed known at each mesh point at a particular time, a Dufort-Frankel [42] difference equation approach was utilized to solve the vorticity transport equation. To handle non-linear advection terms, a special difference form was

applied which conserved vorticity and kinetic energy, thus preventing nonlinear instability.

iii) The vorticity function, in terms of the stream function, was solved by relaxation, using the Liebmann [37] procedure to obtain new stream function values.

iv) The stream function values were then used to obtain new velocity components.

v) New velocity and vorticity values were substituted into the vorticity-transport equation and the process was repeated until a steady state was obtained.

In their original studies, simulated motion of air past an oblong obstacle that was involved in bomb blast studies produced a streamline plot. A quite different streamline plot is formed if the air mass, rather than the obstacle, is stationary with respect to the observer. Thus, streamline and also isotherm plots are possible.

Although results of the iteration procedure were impressive, Pearson [43] found the wall boundary condition led to instability within the flow field. It also appears that the Dufort-Frankel process for solving the vorticity transport equation has no true advantage over equivalent processes. Computation time is not minimum due to non-usage of the over-relaxation technique. Also, the use of periodic boundary conditions is artificial.

Pearson, and also Esch [44], of the same laboratory, formed improvements in the procedure by solving the vorticity-transport equation using the Peaceman-Rachford method [45]. Over-relaxation was also used to speed computations. Pearson's and Esch's analyses of the finite-difference equation, excluding the nonlinear advection terms, have apparently shown that the time increment requirement needs only be not so large as to

obscure the transients of the solution. A stability analysis, performed so as to include nonlinear terms, also appears to bear out this result. Weber [46] of the University of Utah agreed with Pearson's and Esch's conclusions concerning stability only so far as the Reynolds number for flow was small. Thought was given to decreasing the time steps to prevent divergence at higher Reynolds numbers (a similarity parameter useful in fluid flow studies).

Pearson and Weber, to some extent, each attempted to modify the iteration technique as well as to study the truncation error; the truncation error being the error inherent in the use of integration formulae, that is, the difference between the true and approximate solutions of a differential equation due to one step in the numerical integration process. These errors are bound to accumulate in step-by-step procedures since the starting point at each is approximate and each succeeding step compounds the error. Extensive calculations of concern to the problem of truncation and of concern to the choices of smoothing parameters and convergence criteria are being studied in our own investigations. In comparing Weber's computer program modifications with Pearson's, it appears that the former's method is more accurate. Vorticity and stream function iteration type errors appear to be smaller by several factors. There may be some question, however, as to equal accuracy where higher Reynolds numbers are concerned.

Based upon the above arguments, the approach of Pearson, modified by Esch and extended by Weber was attempted. In turn, programming subroutines were added to allow results for our particular problem, as well as to allow Gerber plot output and graphics console capability. The solution space was now divided into a computational mesh as shown in figure 30.

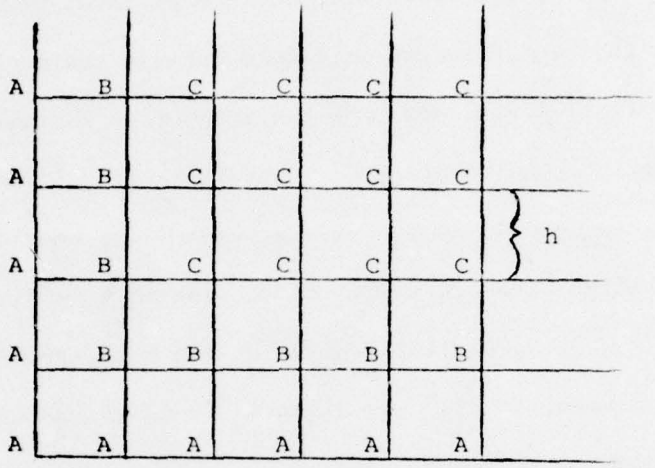


Fig. 30: Computational Mesh Form for the Pearson-Esch Approach.

These mesh forms included unequal stencil arms as have been seen in figures 25 and 29 so as to compensate for eventual employment of curved boundaries. Group A nodes consisted of boundary points, group B were the adjacent points to the boundary, and group C consisted of all other interior points. The mesh cells were declared to be

$$(x_i, y_j, t_k) = (ih, jh, k\,\Delta t) \tag{39}$$

where $i = 1,2,\ldots\ldots M$

$j = 1,2\ldots\ldots N$

$k = 1,2\ldots\ldots$

$$h = \Delta x = \Delta y = 1/(N-1) \tag{40}$$

The discrete stream and vorticity functions $\psi_{i,j}^{k}$ and $\omega_{i,j}^{k}$ were first approximated by $\psi(x_i,y_j,t_k)$ and $\omega(x_i,y_i,t_k)$. It was assumed that all quantities were known at time k and could be calculated for a time of k+1. The procedure then became the following:

i) Superimpose the description of the physical boundaries on the finite-difference grid to determine the mesh lengths a_i, i = 1,2,3,4, and type the mesh nodes (interior, adjacent interior, and boundary). The values of $\psi_{i,j}^{k+1}$ on A, which will be denoted as ψ_{A}^{k+1}, are taken to be equal to the specified boundary values of ψ. The same is true for ω_{A}^{k+1} when vorticity is given on the boundary. On the other hand, if $\partial\psi/\partial n$ is given ω_{A}^{k+1} is an unknown of the problem.

ii) Estimate ψ_{B+C}^{k+1} and ω_{A+B+C}^{k+1} by quadratic extr polation from past values. The formulas are

$$\psi_{i,j}^{k+1} = 3\psi_{i,j}^{k} - 3\psi_{i,j}^{k-1} + \psi_{i,j}^{k-2} \tag{41}$$

$$\omega_{i,j}^{k+1} = 3\omega_{i,j}^{k} - 3\omega_{i,j}^{k-1} + \omega_{i,j}^{k-2} \tag{42}$$

It is seen that this involves storage of three consecutive time levels of vorticity and stream function values.

iii) Improved values of ω_{A}^{k+1} are formed by employing the formula

$$\left.\omega_{i,j}^{k+1}\right|_{\text{improved}} = \text{smoothing factor} \cdot \omega_{i,j}^{k+1} + (1-\text{smoothing factor}) \cdot \left[\frac{-\tau}{2h^2}\right] \tag{43}$$

where $$\tau = 4\psi_{2,j}^{k+1} - 1/2\psi_{3,j}^{k+1} + \psi_{1,j+1}^{k+1} + \psi_{1,j-1}^{k+1} - 5.5\psi_{1,j}^{k+1} - 3h\left(\frac{\partial\psi}{\partial x}\right)_{1,j}^{k+1} \tag{44}$$

The smoothing factor is necessary to prevent overcorrection and subsequent solution divergence. This formula is quickly derived by taking an x=0 and i=1 boundary as an example and writing

$$\psi_{2,j} = \psi + h\frac{\partial\psi}{\partial x} + h^2\frac{\partial^2\psi}{\partial x^2} + \frac{h^3}{6}\frac{\partial^3\psi}{\partial x^3} + O(h^4) \qquad (45)$$

$$\psi_{3,j} = \psi + 2h\frac{\partial\psi}{\partial x} + \frac{4h^2}{2}\frac{\partial^2\psi}{\partial x^2} + \frac{3h^3}{6}\frac{\partial^2\psi}{\partial x^3} + O(h^4) \qquad (46)$$

where all quantities on the right are evaluated at (1,j). Eliminating $\partial^3\psi/\partial x^3$ and then solving for $\partial^2\psi/\partial x^2$ one obtains

$$h^2\left(\frac{\partial^2\psi}{\partial x^2}\right)_{1,j} = 4\psi_{2,j} - 1/2\psi_{3,j} - 3.5\psi_{1,j} - 3h\left(\frac{\partial\psi}{\partial x}\right)_{1,j} + O(h^4) \qquad (47)$$

By employing this result, together with the usual formula for $h^2 \cdot \partial^2\psi/\partial x^2$ in the Poisson's equation $\nabla^2\psi = -2\omega$, one obtains equation (43).

iv) The vorticity transport equation is then solved for ω_{B+C}^{k+1} by using the Peaceman-Rachford method. Recalling that the working equations for the stream function and vorticity in finite difference form were shown in equations (36) and (38), it is then required in this approach that the time factor ($\partial/\partial t$) must be accounted for, at this point. This is accomplished by sweeping across the rows of nodal points in the mesh (the array is a matrix equivalent as stored in the computer) to solve for $\omega_{i,j}$ at new time values. For the vorticity transport equation;

$$\left.\frac{\partial\omega}{\partial t}\right|_{t=t_o+\Delta t/2} = \left.\frac{\partial^2\omega}{\partial x^2}\right|_{t_o+\Delta t/2} + \left.\frac{\partial^2\omega}{\partial y^2}\right|_{t_o}$$

$$+\left[\left.\frac{\partial\psi}{\partial x}\right|_{t_o} \left.\frac{\partial\omega}{\partial y}\right|_{t_o}\right] - \left[\left.\frac{\partial\psi}{\partial y}\right|_{t_o+\Delta t/2} \left.\frac{\partial\omega}{\partial x}\right|_{t_o+\Delta t/2}\right] \qquad (48)$$

This is followed by sweeping by columns;

$$\left.\frac{\partial\omega}{\partial t}\right|_{t=t_o+\Delta t} = \left.\frac{\partial^2\omega}{\partial x^2}\right|_{t_o+\Delta t/2} + \left.\frac{\partial^2\omega}{\partial y^2}\right|_{t_o+\Delta t}$$

$$+\left[\left.\frac{\partial\psi}{\partial x}\right|_{t_o+\Delta t} \left.\frac{\partial\omega}{\partial y}\right|_{t_o+\Delta t}\right] - \left[\left.\frac{\partial\psi}{\partial y}\right|_{t_o+\Delta t/2} \left.\frac{\partial\omega}{\partial x}\right|_{t_o+\Delta t/2}\right] \qquad (49)$$

Again, the procedure is followed for the stream function relationship;

$$\left.\nabla^2\psi\right|_{t_o+\Delta t} = -2\omega_{t_o+\Delta t} \qquad (50)$$

The computer stored matrix has values of the quantity $\omega^*_{i,j}$ as defined by

$$\omega^*_{i,j} \equiv .5\left[\omega^k_{ij} + \omega^{k+1}_{i,j}\right] \qquad (51)$$

where also

$$\psi^*_{i,j} \equiv .5\left[\psi^{k+1}_{i,j} + \psi^k_{i,j}\right] \qquad (52)$$

One should note that if the stencil arms are equal for each cell, as they were for the first set of plots formed, (a_i, i = 1,2,3,4 = unity), then with the application of the second central difference, δ^2_x,

$$\delta^2_x \omega_{i,j} \equiv \omega_{i+1,j} - 2\omega_{i,j} + \omega_{i-1,j} \qquad (53)$$

the required finite difference formulae become

$$\frac{\omega^{*}_{i,j} - \omega^{k}_{i,j}}{.5\Delta t} = \frac{\delta^2_x \omega^{*}_{i,j}}{h^2} + \frac{\delta^2_y \omega^{k}_{i,j}}{h^2} + \frac{1}{4h^2}\left[\left(\omega^{k}_{i,j+1} - \omega^{k}_{i,j-1}\right)\left(\psi^{k}_{i+1,j} - \psi^{k}_{i-1,j}\right) - \left(\omega^{*}_{i+1,j} - \omega^{*}_{i-1,j}\right)\left(\psi^{*}_{i,j+1} - \psi^{*}_{i,j-1}\right)\right] \tag{54}$$

$$\omega^{k+1}_{i,j} - \omega^{*}_{i,j} = \frac{\delta^2_x \omega^{*}_{i,j}}{h^2} + \frac{\delta^2_y \omega^{k+1}_{i,j}}{h^2} + \frac{1}{4h^2}\left[\left(\omega^{k+1}_{i,j+1} - \omega^{k+1}_{i,j-1}\right)\left(\psi^{k+1}_{i+1,j} - \psi^{k+1}_{i-1,j}\right) - \left(\omega^{*}_{i+1,j} - \omega^{*}_{i-1,j}\right)\left(\psi^{*}_{i,j+1} - \psi^{*}_{i,j-1}\right)\right] \tag{55}$$

where $\psi^{*}_{i,j}$ uses the most recent estimate of $\psi^{k+1}_{i,j}$ from step v of the procedure. These equations yield a tri-diagonal coefficient matrix which may be solved according to an algorithm by Todd [47].

v) From the nondimensional form of the vorticity function, improved values for ψ^{k+1}_{B+C} are calculated. To accomplish this step it is necessary to employ successive over-relaxation [48] and the use of ψ^{k+1}_{A} as boundary values. It should be noticed that the source term in the Poisson equation changes, using the results of the previous step.

vi) The process is repeated by returning to step iii and the iterations continue until the stream function and vorticity values reach a prescribed maximum change tolerance. The tolerance, as set by Weber, are functions of the Reynolds number. Note that this is because he used an absolute error test rather than the more useful relative error test. Values

of the stream function and vorticity in the dimensionless forms in the Weber approach increase with increasing Reynolds number. The error may be expected to increase also.

As to stability criteria it appeared to Pearson and Esch that no criteria problem was presented as long as the flow was restricted to Reynolds numbers less than a value of 100. Weber's view of stability requirements appear to be correct, however, for higher Reynolds numbers. For Reynolds numbers in the range of 3000 the time step of .005 allowed convergence if the mesh size was .1. Recalling that the dimensionless function g(t), used to form the velocity change with time on the boundary, was $1-e^{-at}$, alternately one could restrict

$$(1 - e^{-20t}) \cdot (R_{e_{new}} - R_{e_{old}}) < 30 \qquad (56)$$

for non-divergence. In turn, convergence tolerances and truncation error computation followed the approach of Weber*.

5. Boundary Conditions.

Since the stream function is specified as a boundary condition, ψ and either its normal derivative ψ_n or the vorticity ω must also be specified. The normal derivative was specified with the stream function at both walls and at the inflow and outflow region sections since physically, this was equivalent to specifying the u and v velocity components on the boundaries.

---

* The investigation formed by Weber included the aid of Mr. Charles Brauer in forming an amount of program subroutines equivalent to certain numerical procedures. In turn, Mr. Brauer formed like subroutines for this investigation, thus allowing the Weber and Brauer views to be similar in both projects.

With the assumption of channel flow and parallel solid walls for the upper and lower boundaries of the solution space,

$$\psi\ (x,0) = 0$$

$$\psi\ (x,1) = R_e f(t)$$

$$\psi_y(x,1) = \psi_y\ (x,0) = u_{wall} = 0$$

$$\psi_x(x,0) = \psi_x\ (x,1) = -v_{wall} = 0 \qquad (57)$$

i.e., the upper and lower walls take on the value of a stream line and the velocity at the walls is zero. Along the left or inflow boundary a choice is available for the desired entrance velocity profile. It is to be noted that rather than form the normal derivative at the inflow and outflow boundaries as first proposed by Pearson, Esch utilized the vorticity with the stream function. In turn, by specifying the vorticities to be equal to zero at the entrance channel, the generation of unrealistic vorticites at that position, as is sometimes seen with the Pearson approach, was avoided. However, one could question this point. Hence, the entrance profile

$$\psi\ (0,y) = yR_e \qquad 0 \leq y \leq 1$$

and thus

$$\psi_y(0,y) = u(0,y) = R_e$$

$$\psi_x(0,y) = -v(0,y) = 0 \qquad (58)$$

i.e., at first a rectangular velocity distribution was specified at the entrance to the channel. However, the velocity profile of an incompressible viscous fluid, as it flows between two parallel plates gradually changes, as it develops fully from the initial profile to the classical parabolic profile, seen in Poisouille flow. But, regardless of the shape of the duct's cross section no exact solution is had for the profile because of the nonlinear advection terms in the equations of motion of the problem. Certain approximate analytic solutions have been tried by various investigators using either a series approximation or a division of the entrance

region into two zones. Although there is more recent work the method formed by Schlichting [49] is basic to subsequent studies in that at the zone near the channel entrance, the boundary layer equations are applied and an approximate solution is obtained, in terms of a perturbation series. In the zone where the profile is almost fully developed, the solution is obtained as a perturbation of the fully developed velocity profile. These two solutions are matched at a suitable point to form a complete solution. Since this problem of evaluating the true velocity profile at the duct entrance region is a difficult one which requires more study, at this point of the continuing investigation, fully developed flow in the form of a parabolic velocity profile at the entrance region, was assumed.

Theoretically, for a conduit with an abrupt expansion, uniform flow will only be obtained at infinity in either direction from the expansion. Our situation is assumed to be a flow into an expansion region, especially when viewing the blood flow as coming through the heart valve seat into the sinus of Valsalvae region, upstream of the obstacle. In turn, the velocity profile of uniform laminar flow is known to be parabolic. Obviously, for a computational solution, the length of the conduit must be as short as is compatible with desired accuracy and limitation of computer storage and time. For a low Reynolds number the velocity profile approaches a parabolic form after a relatively short length from the entrance in a uniform conduit, according to the classical approaches of Prandtl and Tietjens [50] and Schlichting [51]. For our range of Reynolds numbers it was therefore assumed that a parabolic velocity distribution at a proper finite distance upstream, as well as downstream from the entrance region, was reasonable until more detailed investigations could be formed. Even for a distance of 3/4 D_o (D_o being the

diameter of the inlet) upstream, Hung [52] has confirmed that uniform flow at the entry region position would not be disturbed noticeably by any nonuniformity on the downstream side. (Note that this assumption is known to be untrue for Re < 50 and questionable between 50 <Re <250.)

The velocity and stream function profiles of the fluid leaving the channel is, in general, unknown, and moreover, it is desirable not to detail the profiles explicitly. Small changes in the exit velocity profile would produce large effects on the interior stream function and vorticities. In order to allow this profile to develop in a natural fashion rather than to force it to predetermined conditions, the vorticity and stream function at the exit were made equal to their values on the adjoining interior column of grid points. In a rapidly changing velocity field a time extrapolation was also included, based upon changes in the vorticity and stream functions during the preceding two time steps.

In some calculations, some perturbation effects were placed upstream of the obstacle in both a symmetric and asymmetric fashion, however, such attempts were exploratory ones.

C. Display of Computer Graphics Results - Obstacle and Mesh Lines Coincidental.

Every parameter in the computer program was kept constant except the obstacle shape in a particular experiment, in order to determine the effect of the obstacle shape on the development of the flow behind the obstacle.

number of mesh rows: 26

number of mesh columns: 61

Reynolds number: 1000 (unless otherwise stated)

error tolerance:

vorticity function: $0.001\ R_e$

stream function: $0.00003\ R_e$

or

$0.0005\ R_e$

The series of computer program subroutines formed at this junction of the investigation allowed two interesting evaluations. For each obstacle of interest, computer graphics results were developed for the stream function, vorticity function, horizontal and vertical velocity components, total velocity and velocity vector. In turn, all such plots, except the velocity vector, consisted of two parts; a solutior space surface plot and its isometric projection. An additional program subroutine allowed the presentation, by choice, of a particular time frame of any of the above desired functional plots, as formed with the aid of a Gerber plotter. Figure 31 shows an example of a console display result; in this case a stream function.

For the situation of the stream function, such a plot can be verified by a laboratory flow visualization technique such as the injection of a dye or radio-active particles into the fluid. The stream function isometric plot shows the relative magnitude of the stream lines, thus forming a semi-quantitative evaluation. The vorticity function plot indicates the magnitude of the angular momentum of a fluid particle in the solution space. This plot cannot be obtained by laboratory experiment. In the case of the velocity plots the solution space surface plots are equivalent to those obtained experimentally in the laboratory. These are gained by traversing in a column or row direction the space with a hot wire anemometer probe so as to obtain the magnitude of the velocity or desired velocity component. The contour plot forms the

trace of the equivalue line or isovelocity line of flow. Such a plot would be required to move through the solution space in such a manner as to have its reading held at a constant value. The velocity vector plot indicates both the direction and relative magnitude of the flow. This is analogous to the rather crude laboratory method of visualizing the flow direction by inserting "streamers" into the flow stream.

Figure 32 is a comparison of the u-velocity profile for a two dimensional channel flow situation as formed analytically in reference 53. The significance is that only the profile of the flow entering the channel was specified. The computer program calculated its own exit profile. This is important since, in general, an investigator knows only the profile of the fluid that enters the field of interest.

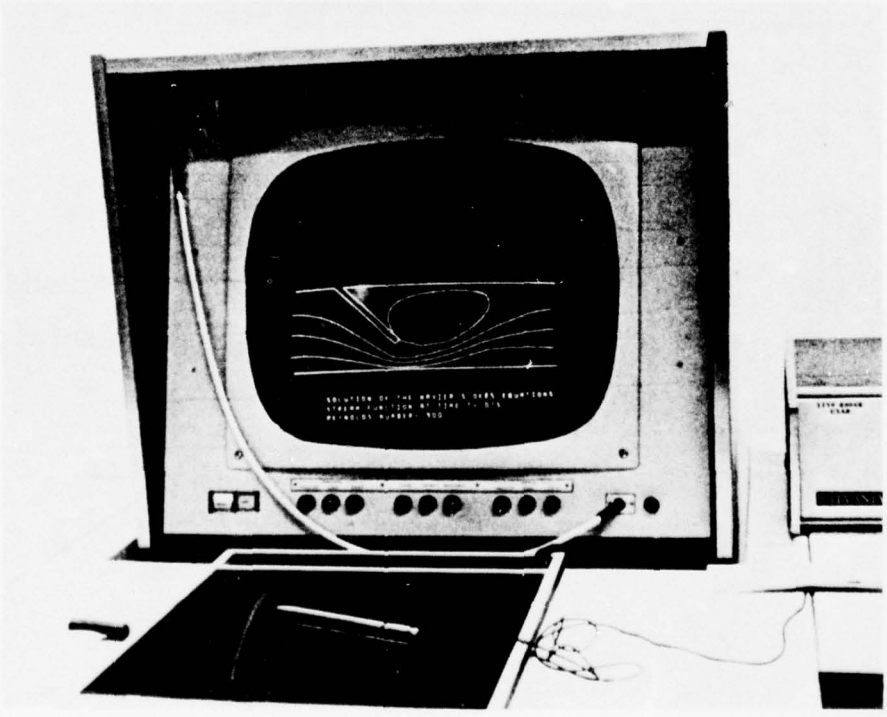


Fig. 31: A Typical Result For Flow About An Obstacle As Seen on the IDI Scope.

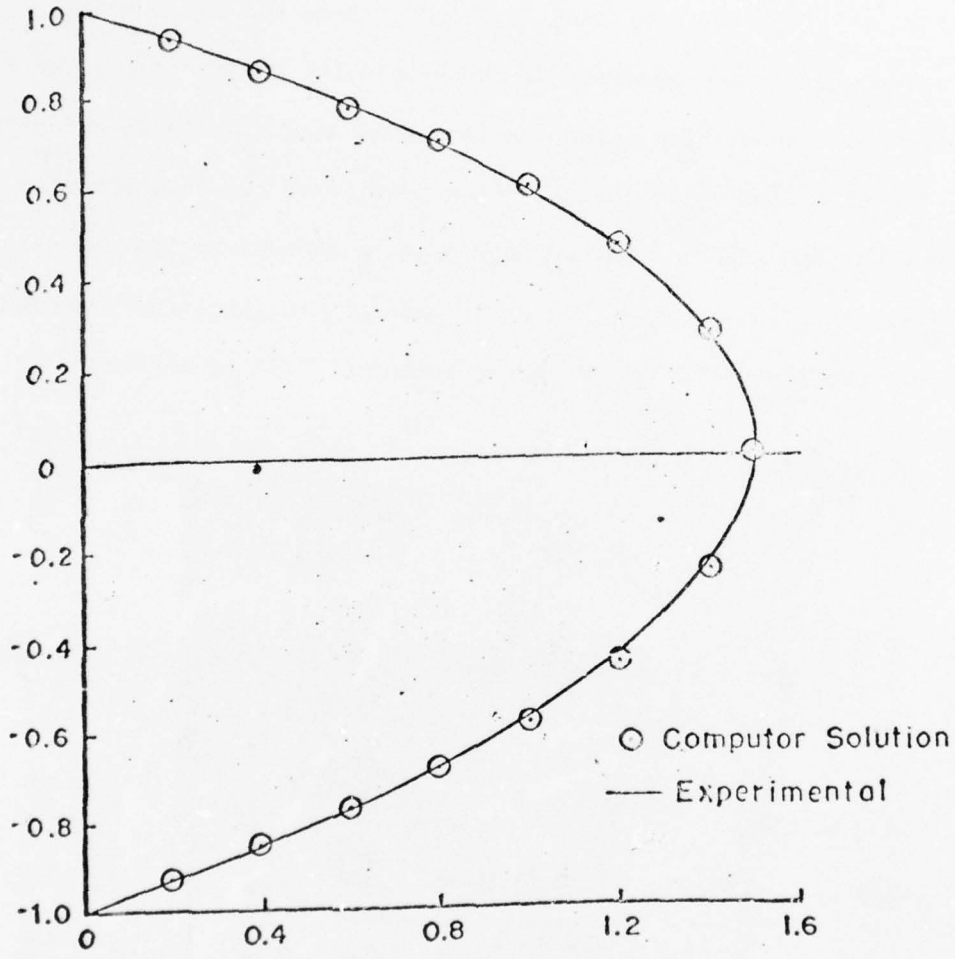
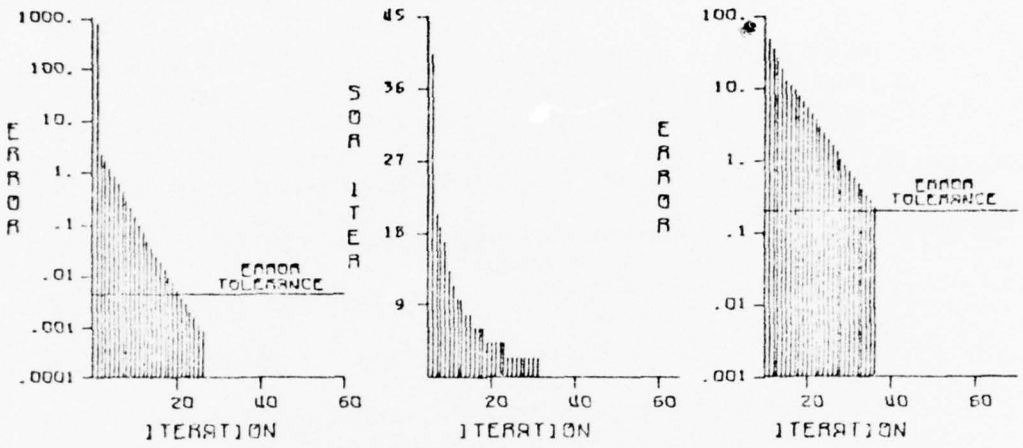
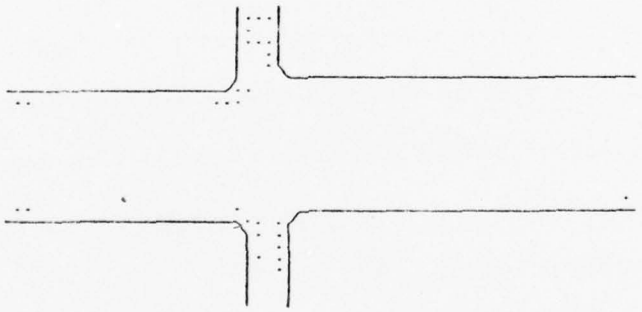


Fig. 32: Comparison of Computer Results with Predicted Parabolic Profile

Figure 33 shows the stability values of the numerical routine for certain stream and vorticity functions and largest error positions, as originally displayed on the IDI display scope for changing iteration values. Figure 33, however, is a hard copy of one time frame within a changing time series seen on the display screen. The top left plot in this particular diagram shows the anount of error versus amount of required iterations performed by the computer for forming a particular stream function exhibit. During each iteration the computer monitored the allowable amount of error during the computational process. When the analysis searcher (computer) arrived at the required error tolerance, it initiated evaluation of the stream function for the next time increment. The middle top plot shows the convergence of the numerical routine to a desired value, accomplished by monitoring the stream function error term. A like error tolerance evaluation is shown for the vorticity function. The bottom view shows positions of the largest error terms for the vorticity function in an idealized orthogonal renal section at a particular instant of analysis. These positions are where the vorticity function changed most rapidly during each iteration, dependent upon parametric values listed below the plot.

The rather large series of computational results on the succeeding pages depict various end products as based upon the previously noted numerical analysis and resulting algorithms. It is to be noted that all results at this point are based upon the requirement that the boundaries of the obstacles and the boundaries of the computing mesh cells are coincidental. Close observation, particularly for the oval and circular obstacles, shows that these objects' boundaries were formed by segmented lines; at the most, any individual line crossing a mesh cell was a diagonal one, allowed by the algorithm. Curved boundaries are discussed in the section that follows this picture group.





LOCATION OF LARGEST ERROR TERMS

		TOTAL(SEC)
FRAME NO.= 1	VORTICITY BOUNDARY=	.08 SEC 2.01
REYNOLDS NUMBER= 50	VORTICITY INTERIOR=	.44 SEC11.60
TOTAL ELAPSED TIME= .73 MIN	STREAM INTERIOR=	.17 SEC 12.02
TIME TO CONVERGE= 25.63 SEC	MISCELANEOUS=	.35 SEC 18.09

Fig. 33: Stability Values of the Computer's Numerical Routine for Certain Stream and Vorticity Functions. Positions of Largest Error Terms for a Vorticity Function with a Particular Reynolds Number Value is also Shown.

Each set of figures for the rectangular (figures 34-48), semi-hemisphere (figures 49-62), circular (figures 63-78) and oval shaped (figures 79-93) obstacles are required for the prosthetic heart valve section of the investigation. Particular stream function frames are first shown to point out the capability of the graphics display for allowing study of a continuously changing flow field (remembering that the present requirement is blood movement downstream of a heart valve occluder or about an atherosclerotic lesion). Results shown are hard copy equivalences to the console display. Subsequently, examples of the u and v and total velocity plots, as well as the velocity vector plot are shown for such obstacle shapes. The velocity vector plot allows the added advantage that if the velocity vector arrows are **not** consistent with their neighbors' paths (including circular motion for the vorticity positions) it is a warning that there may be a computational error at that position. Arbitrary Reynolds number values are shown in each figure.

Another set of figures involve a hexagonal obstacle (figures 94-104) and its functional plots for two different time frames.

As an introduction to the possibility of employment of computer graphics for studying valve leaflet motion, one simplified leaflet form in a steady flow field is shown in figures 105-112 for two different Reynolds numbers and time values. Figures 113-118 show evaluation possibilities for flow from the atrium to the ventricle section of the heart in a very rudimentary shape and fashion, for the present. In all cases one views rigid wall and steady flow conditions. This point will be discussed in a later section. In turn, subroutines to form figures 119-124 are introductions to a two leaflet natural valve study or other problems that require similar shaped sections.

Figures 125-131 again show other usages for computer graphics. In this example, one time frame plots are shown for two obstacles in a flow field.** One last example in this section is the stream function plot of figure 132 which stresses the capability of the procedure as it could pertain to a computer graphics analysis of an arbitrary aircraft wing shape placed at any flow angle.

** This study was evolved by Mr. Charles Brauer for a particular fluidic device study as required by the Fluidonics Corp. of Salt Lake City (now Bio-Logics Inc.)

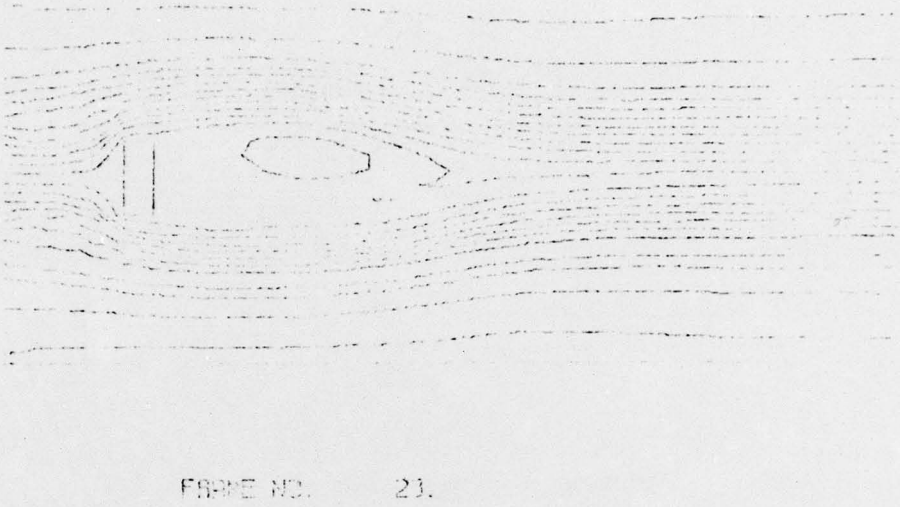


Fig. 34: Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle

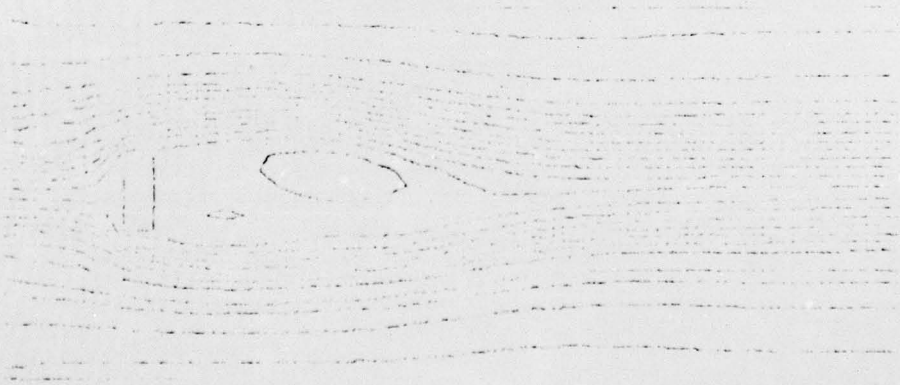


Figure 35: Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle

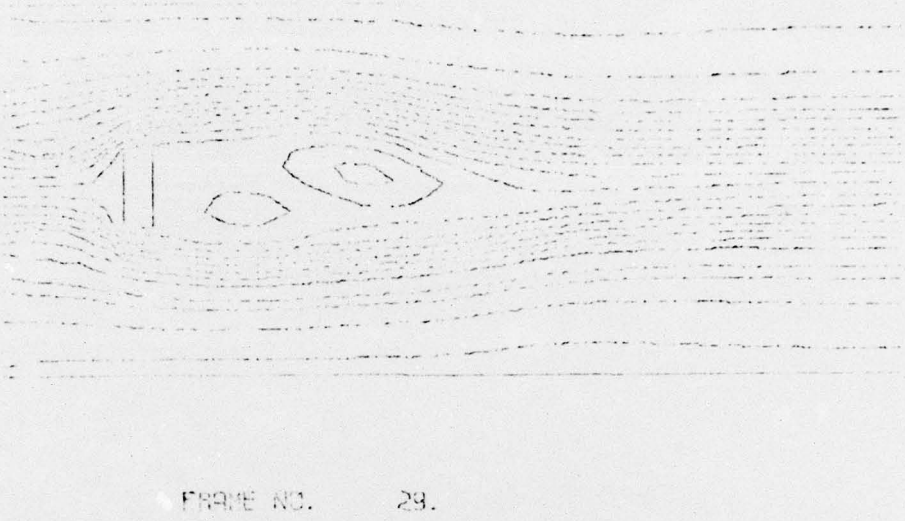


Fig. 36: Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle.

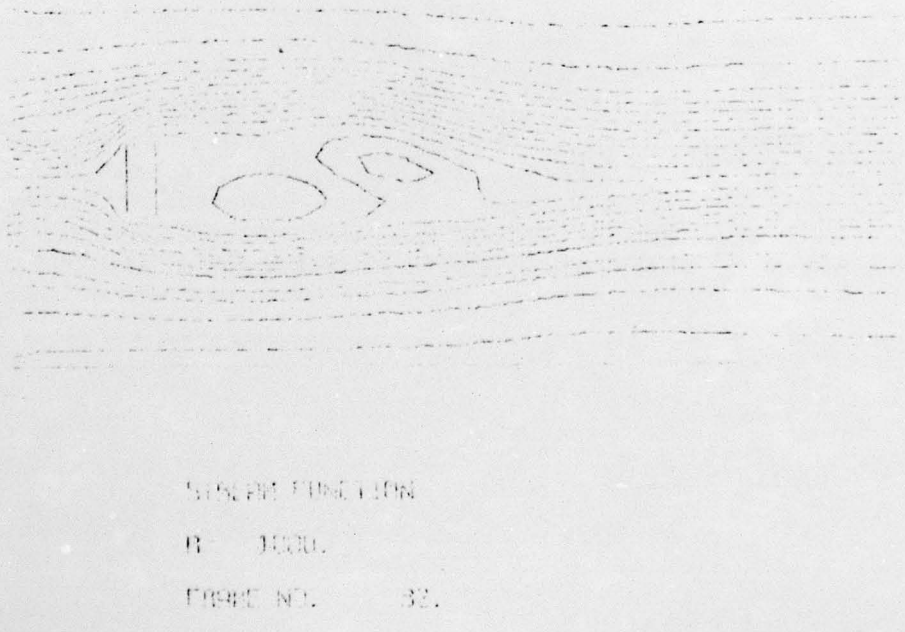


Fig. 37: Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle.

Fig. 38: Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle.

Fig. 39: Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle.

Fig. 40: Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle.

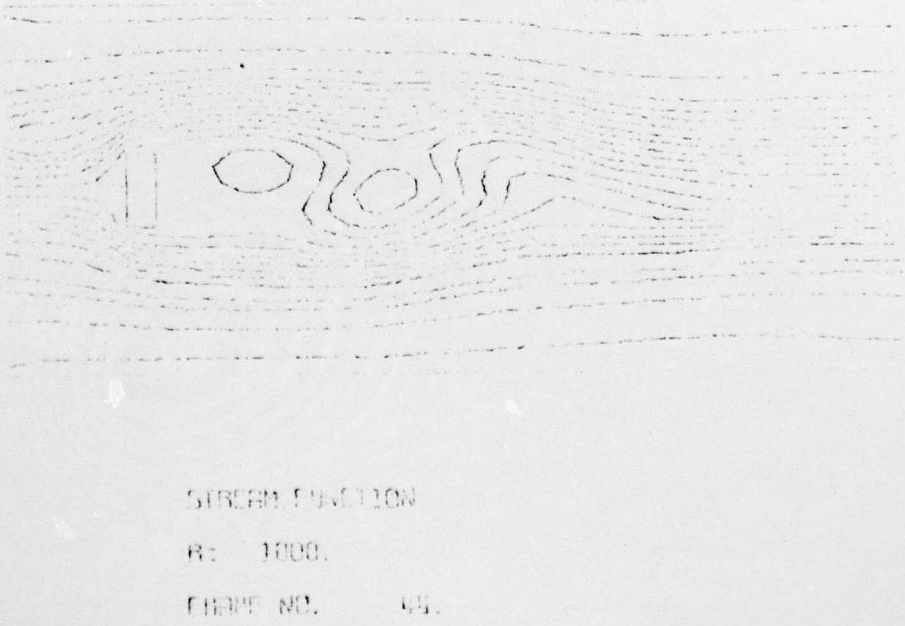


Fig. 41: Stream Function Plot at a Particular Time Frame for a Rectnagular Obstacle.

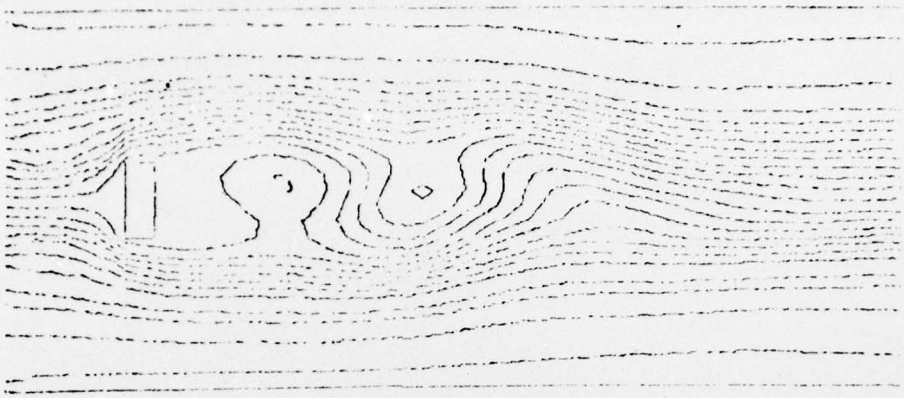


Fig 42: Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle.

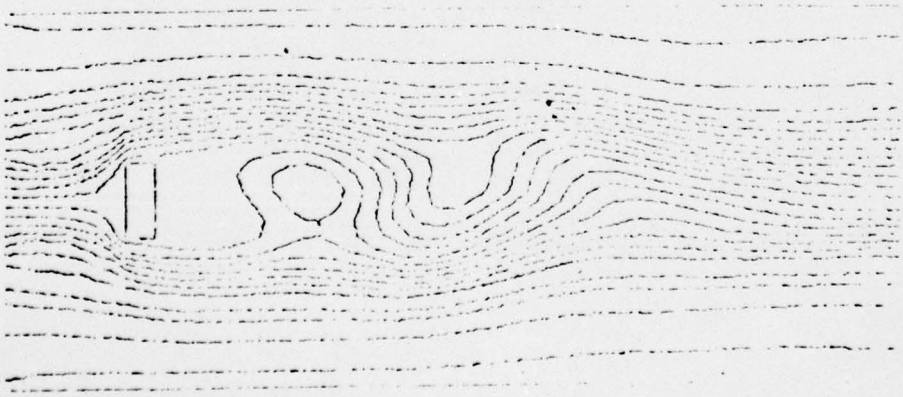


Fig. 43: Stream Function Plot at a Particular Time Frame for a Rectnagular Obstacle.

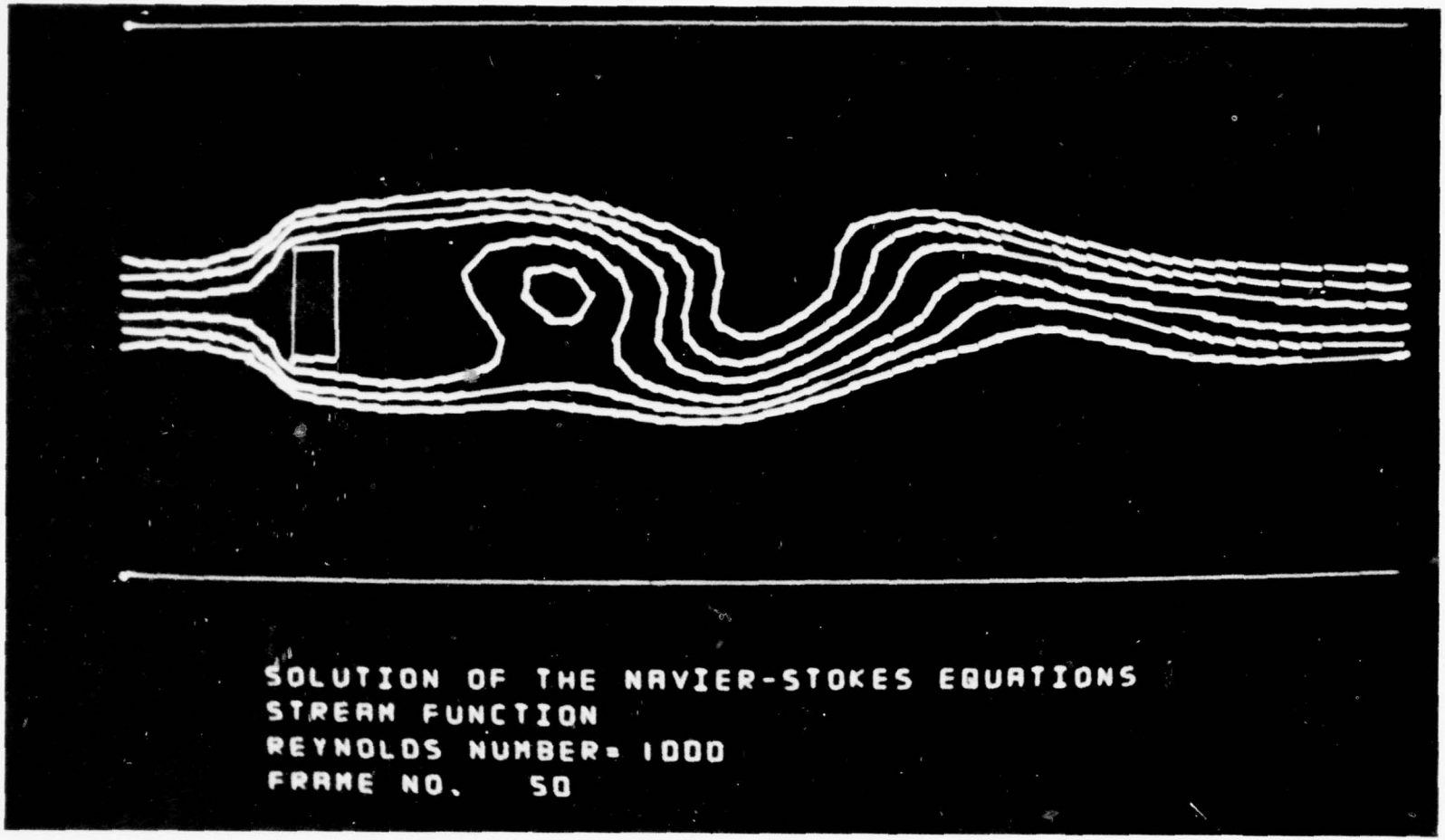


Fig. 44: Stream Function Plot at a Particular Time Frame for a Rectangular Obstacle and For a Higher Reynolds Number Value.

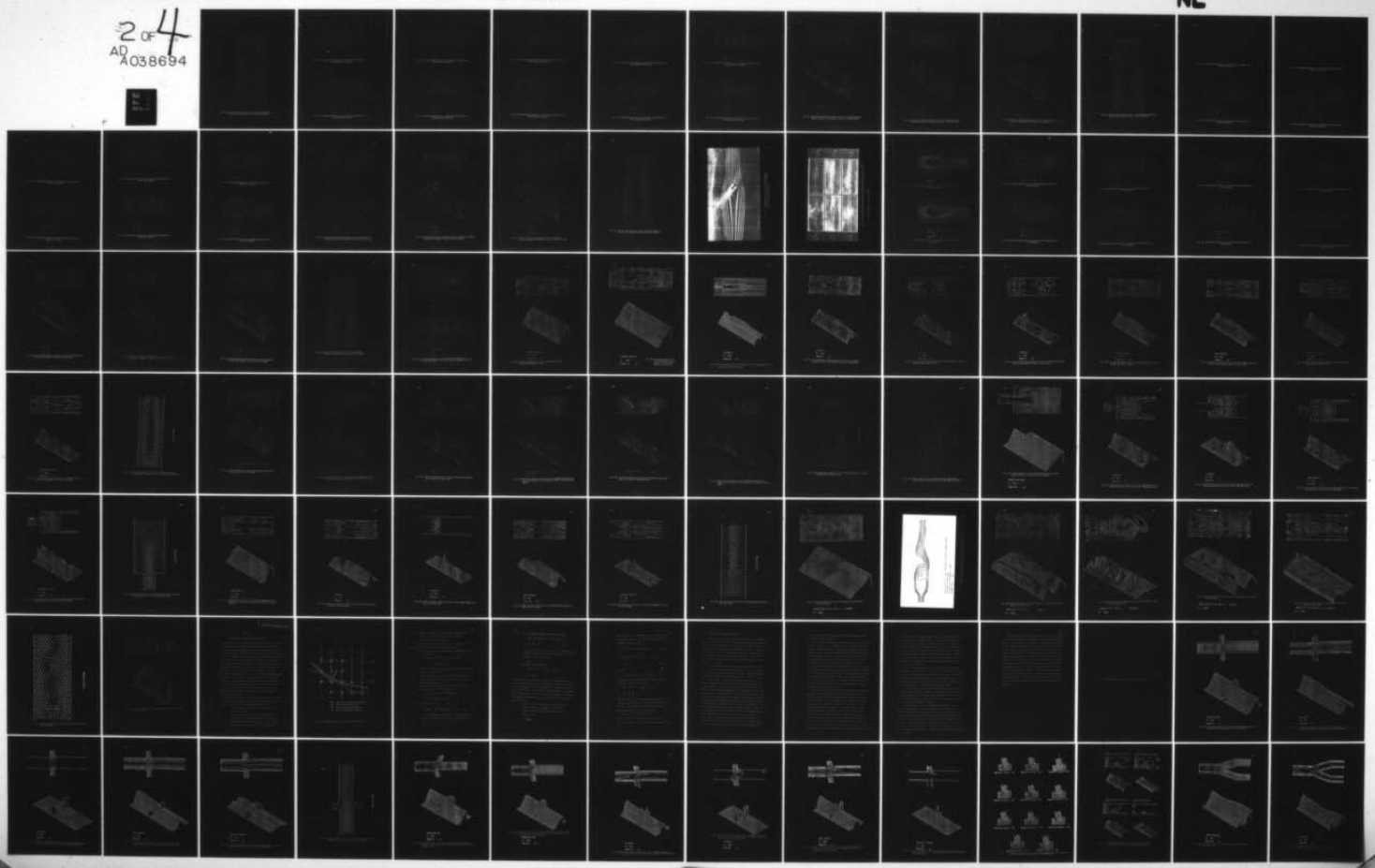
Fig. 45: The Horizontal Velocity Function Plot for a Rectangular Obstacle (Surface and Isometric Views at One Time Frame)

Fig. 46: The Vertical Velocity Function Plot for a Rectangular Obstacle (Surface and Isometric Views at One Time Frame).

Fig. 47: The Total Velocity Function Plot for a Rectangular Obstacle (Surface and Isometric Views at One Time Frame).

AD-A038 694 UTAH UNIV SALT LAKE CITY DEPT OF COMPUTER SCIENCE F/G 6/5
AN APPLICATION OF COMPUTER GRAPHICS: TWO CONCURRENT INVESTIGATI--ETC(U)
NOV 71 H GREENFIELD, R DEBRY
UNCLASSIFIED UTEC-CSC-71-115 F30602-70-C-0300
NL

2 OF 4
AD A038694



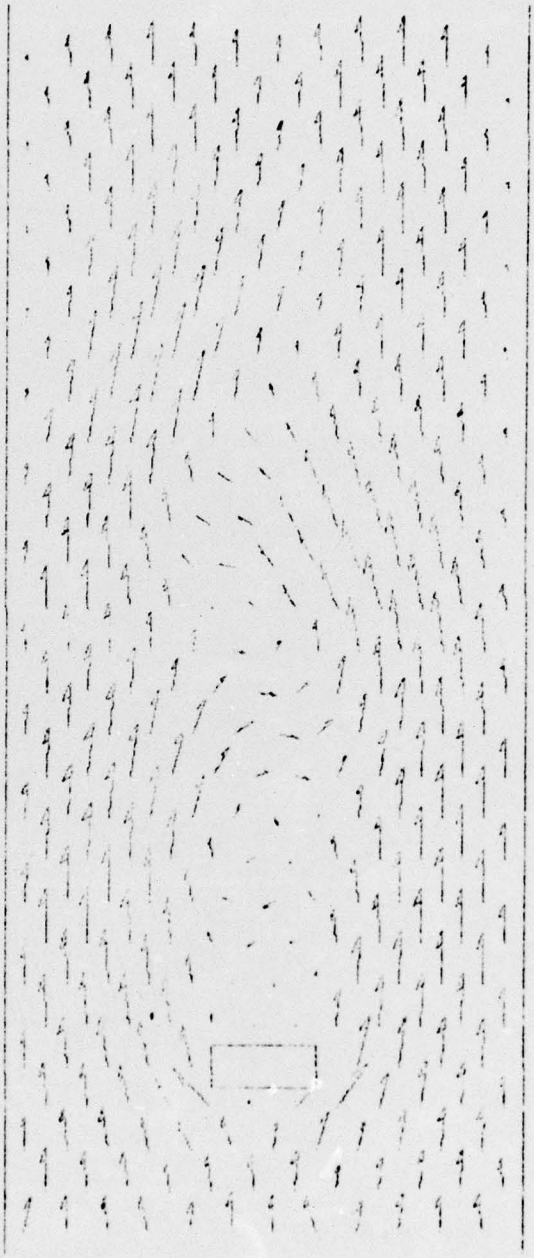


Fig. 48: The Velocity Vector Plot for a Rectangular Obstacle
(Surface and Isometric Views at One Time Frame)

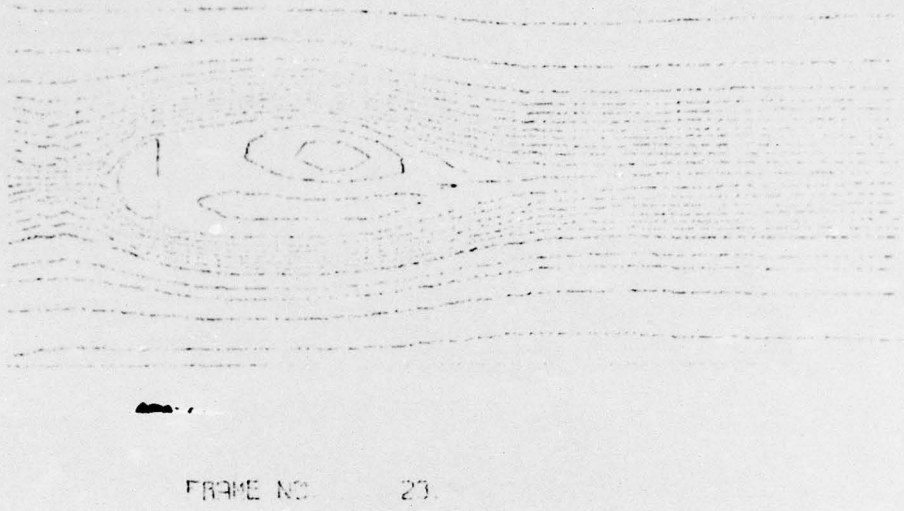


Fig. 49: Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle.

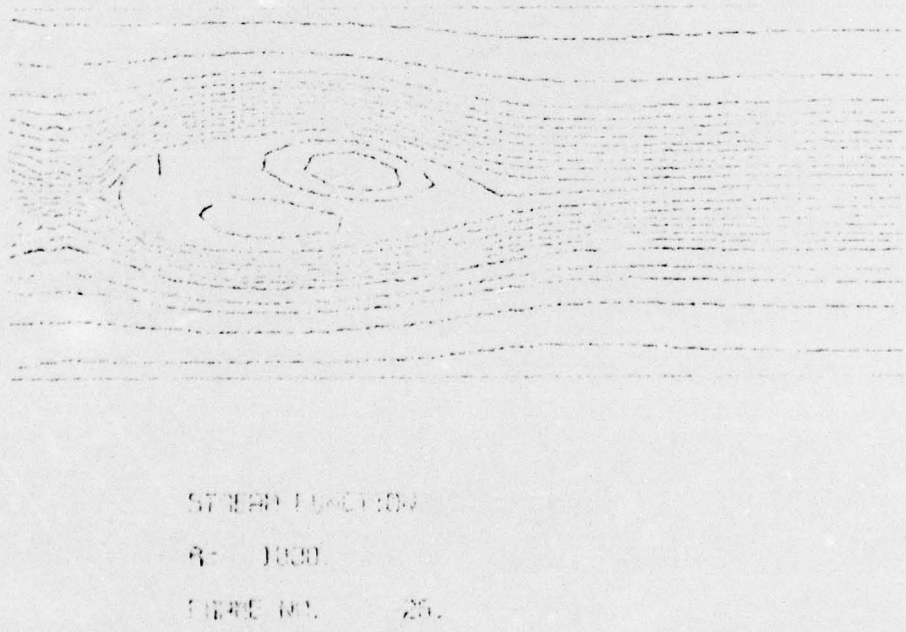


Fig. 50 : Stream Function Plot at a Patticular Time Frame for a Semi-Hemisphere Obstacle.

Fig. 51: Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle.

Fig. 52: Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle.

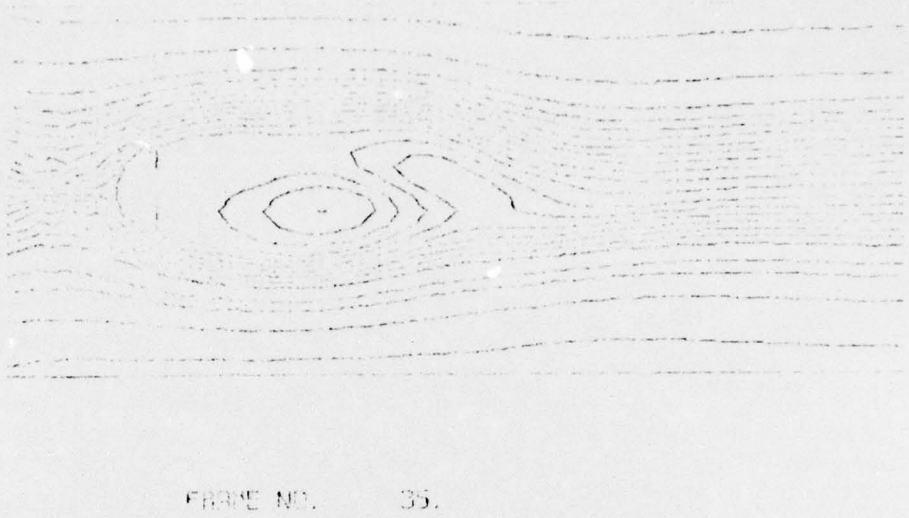


Fig. 53: Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle.

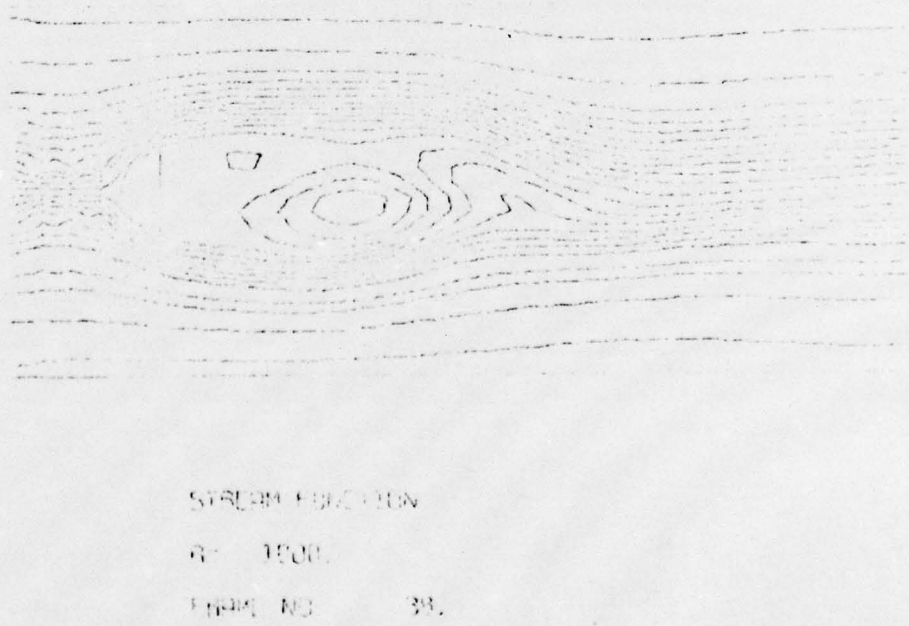


Fig. 54: Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle.

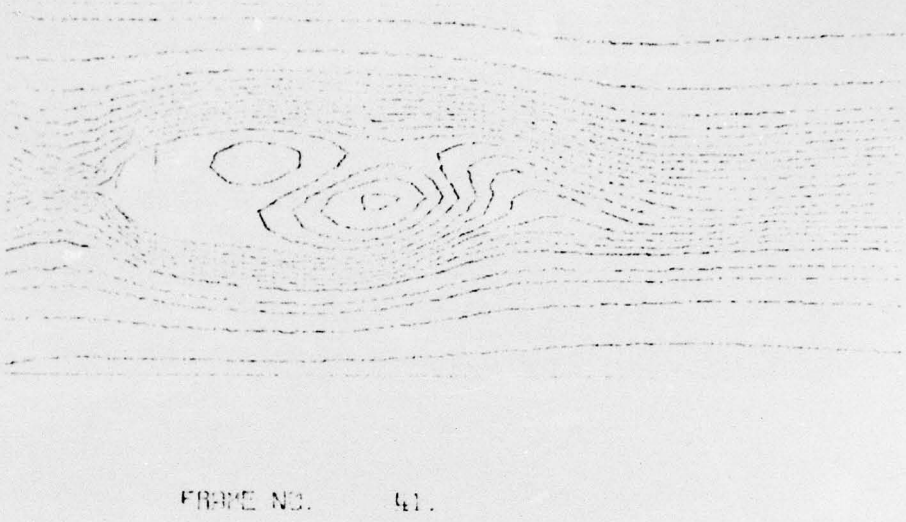


Fig. 55: Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle.

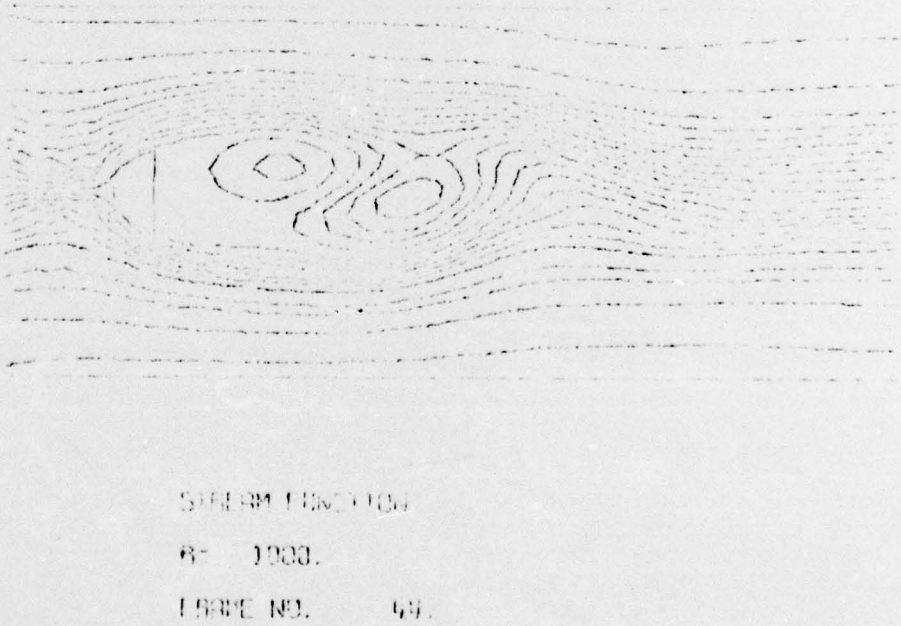


Fig. 56: Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle.

Fig. 57: Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle.

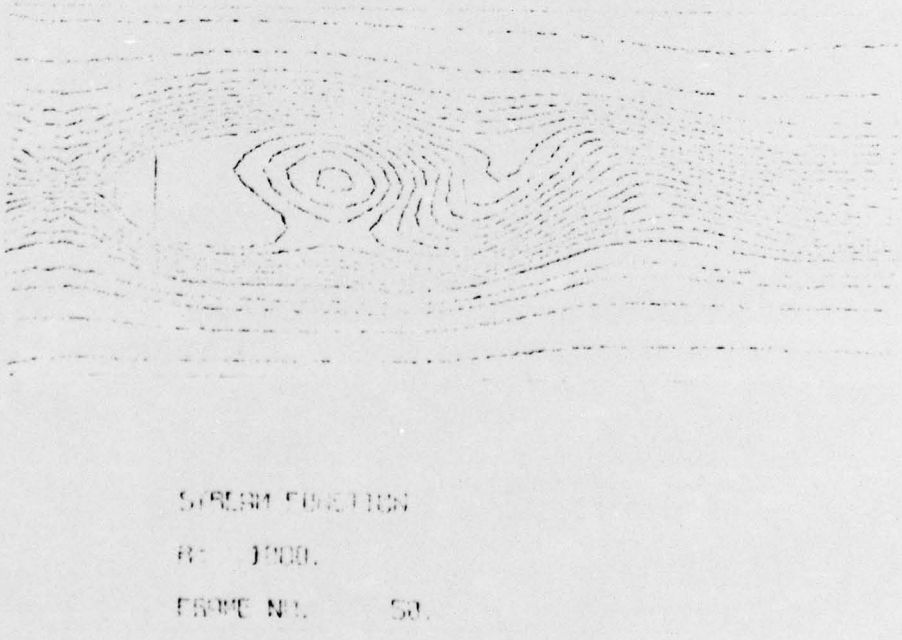
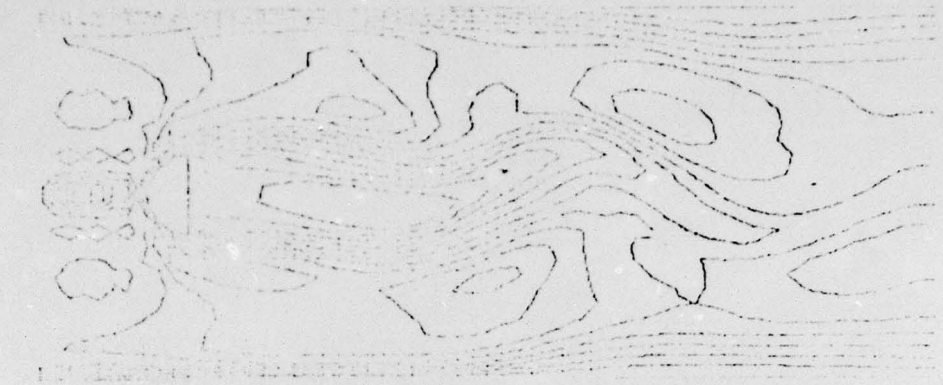


Fig. 58: Stream Function Plot at a Particular Time Frame for a Semi-Hemisphere Obstacle.



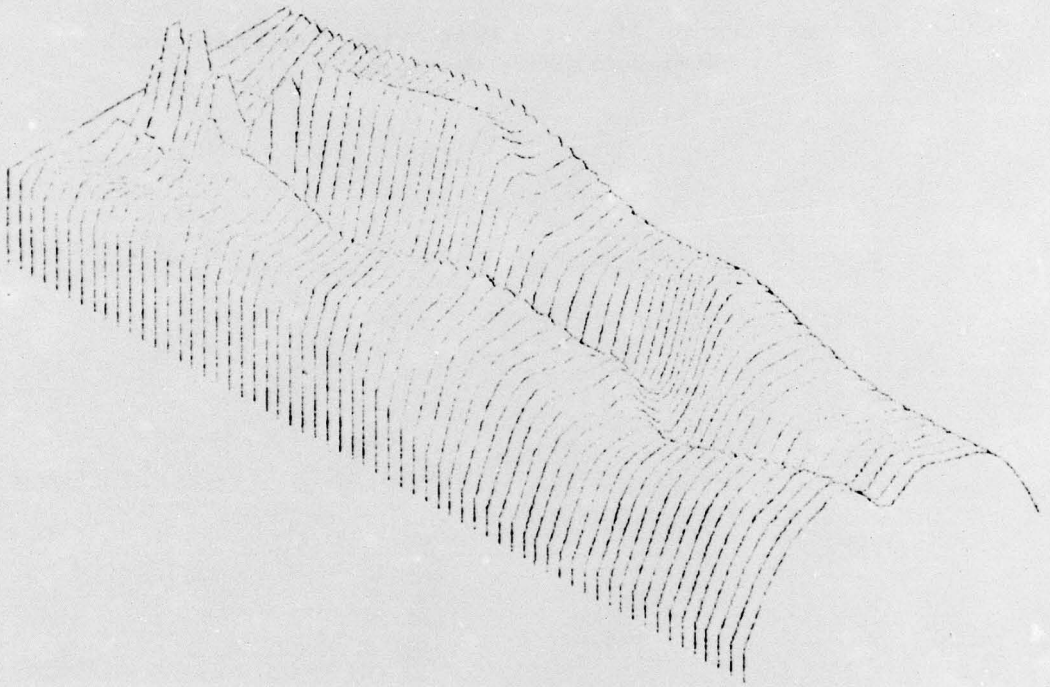


Fig. 59: The Horizontal Velocity Function Plot for a Semi-Hemisphere Obstacle (Surface and Isometric Views at One Time Frame)

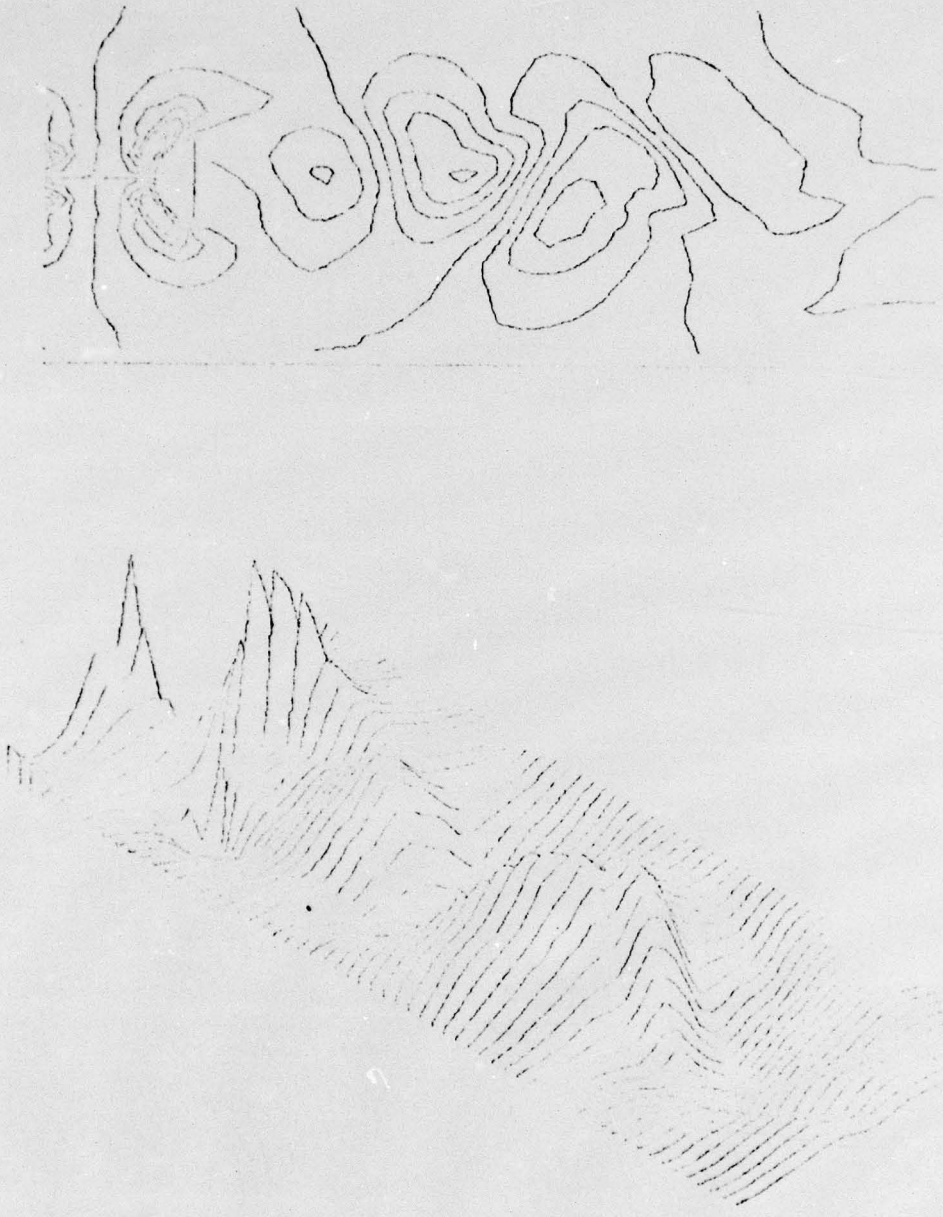


Fig. 60: The Vertical Velocity Function Plot for a Semi-Hemisphere Obstacle (Surface and Isometric Views at One Time Frame)

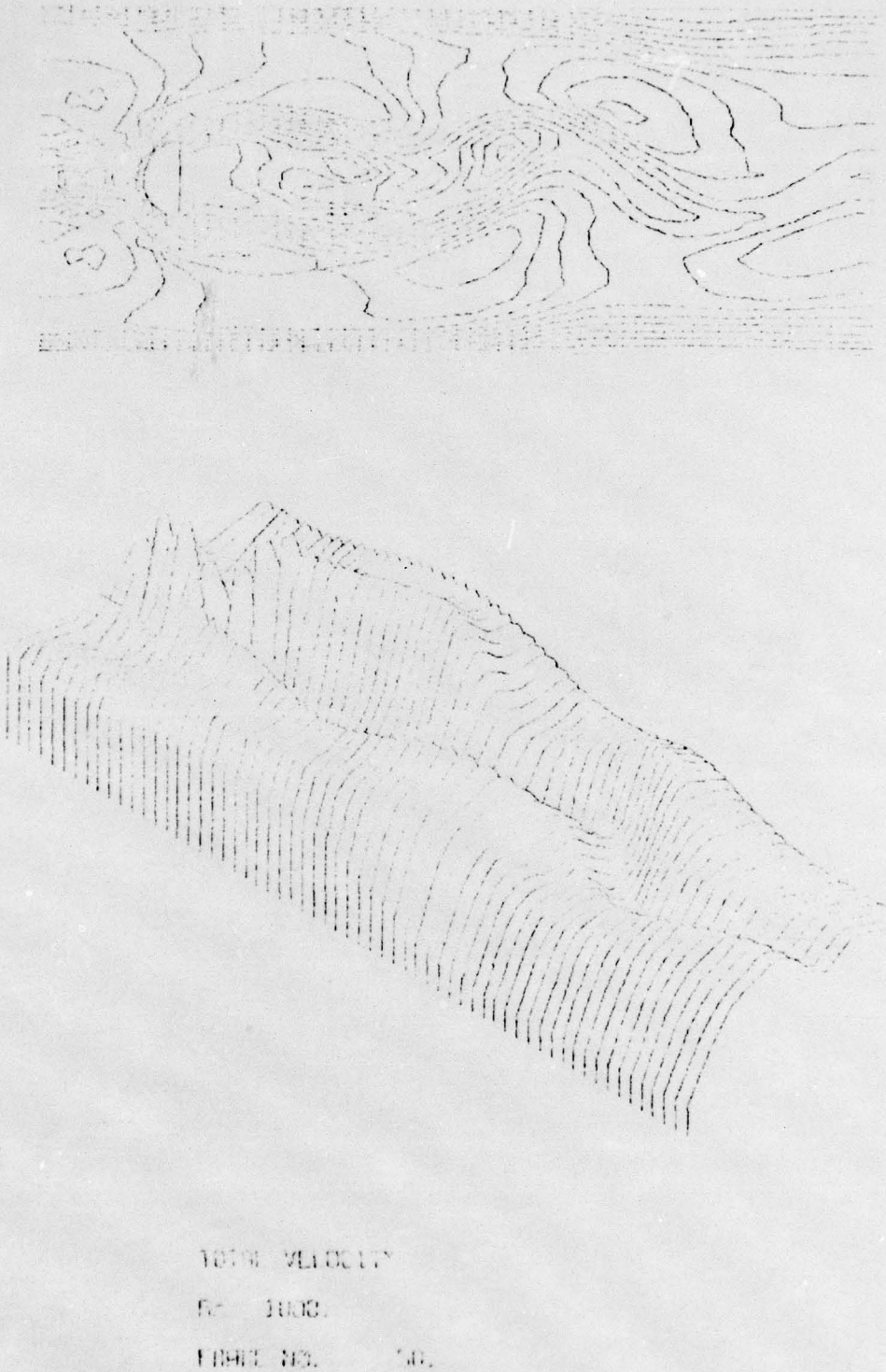


Fig. 61: The Total Velocity Function Plot for a Semi-Hemisphere Obstacle (Surface and Isometric Views at One Time Frame)

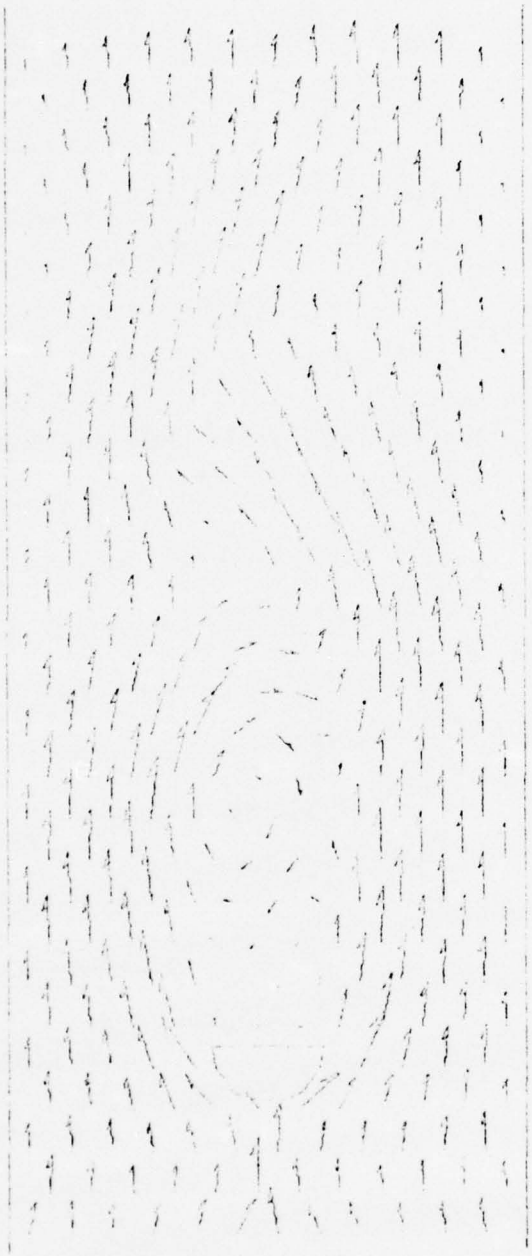


Fig. 62: The Velocity Vector Plot for a Semi-Hemisphere Obstacle
(Surface and Isometric Views at One Time Frame)

Fig. 63: Stream Function Plot at a Particular Time Frame for a Spherical Obstacle.

Fig. 64: Stream Function Plot at a Particular Time Frame for a Spherical Obstacle.

Fig. 65: Stream Function Plot at a Particular Time Frame for a Spherical Obstacle.

Fig. 66: Stream Function Plot at a Particular Time Frame for a Spherical Obstacle.

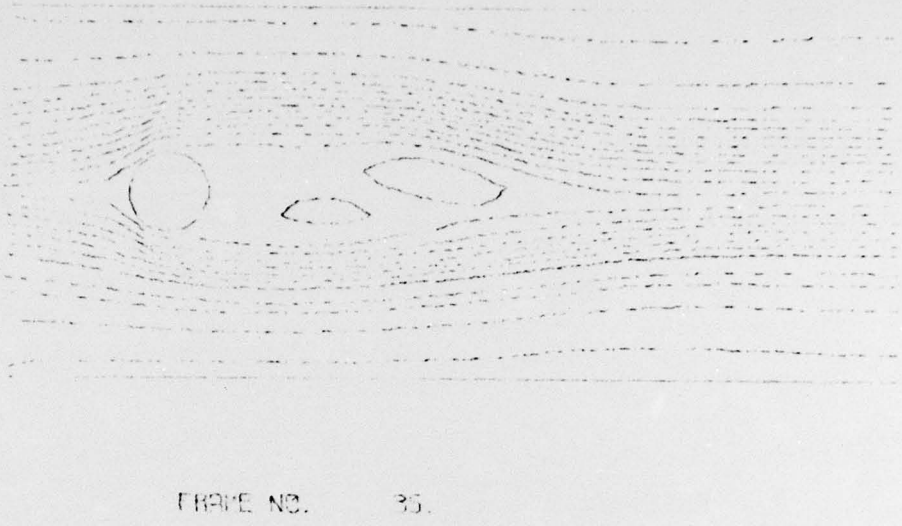


Fig. 67: Stream Function Plot at a Particular Time Frame for a Spherical Obstacle.

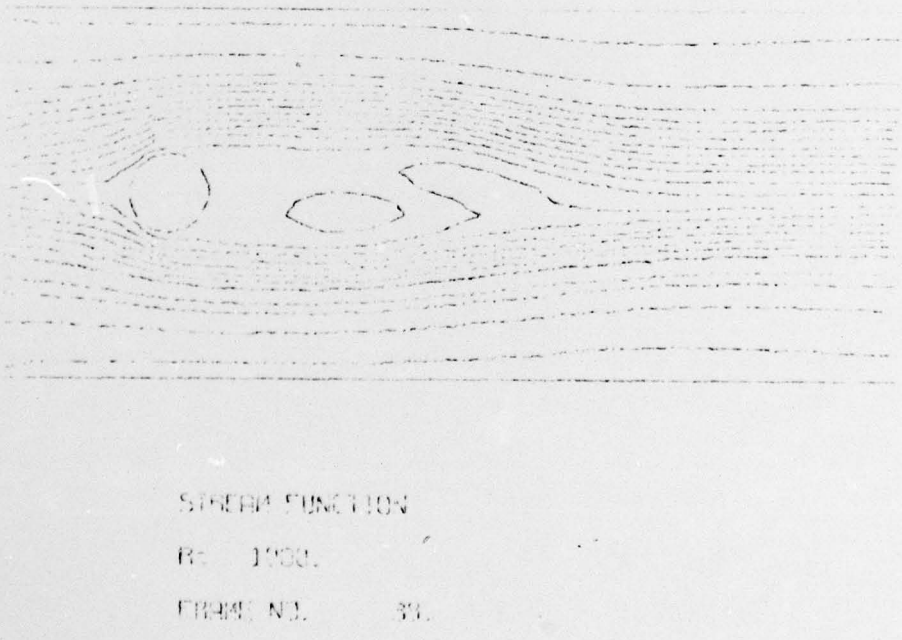


Fig. 68: Stream Function Plot at a Particular Time Frame for a Spherical Obstacle.

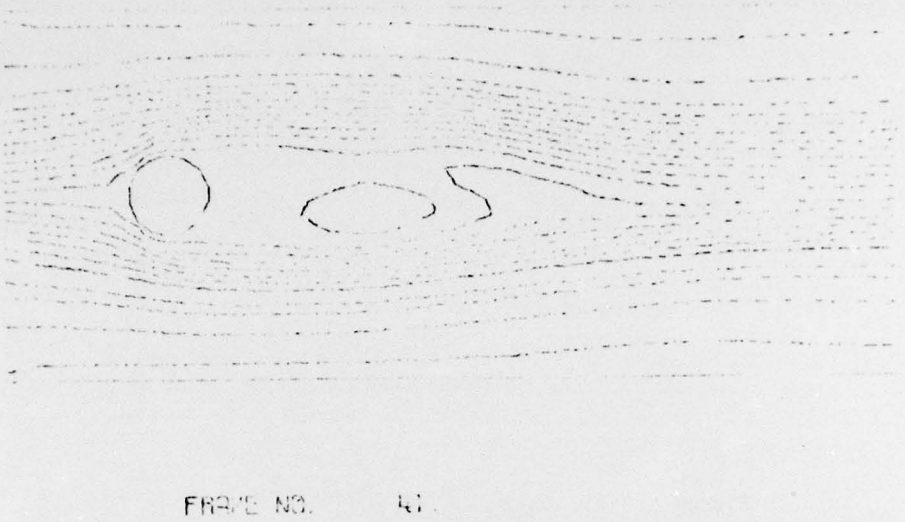


Fig. 69: Stream Function Plot at a Particular Time Frame for a Spherical Obstacle.

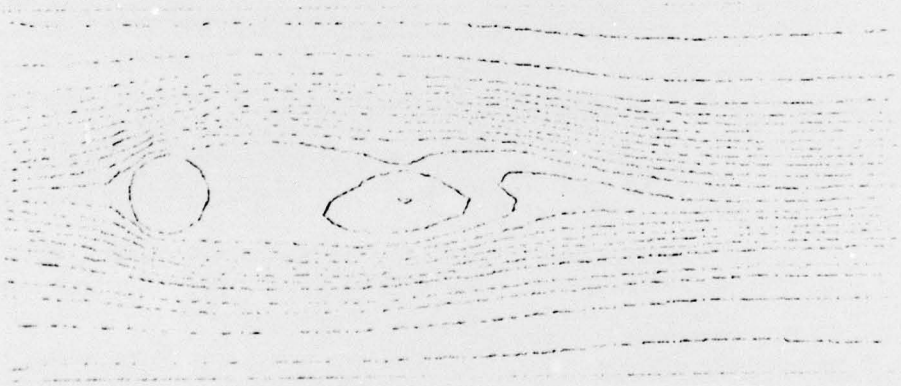


Fig. 70: Stream Function Plot at a Particular Time Frame for a Spherical Obstacle.

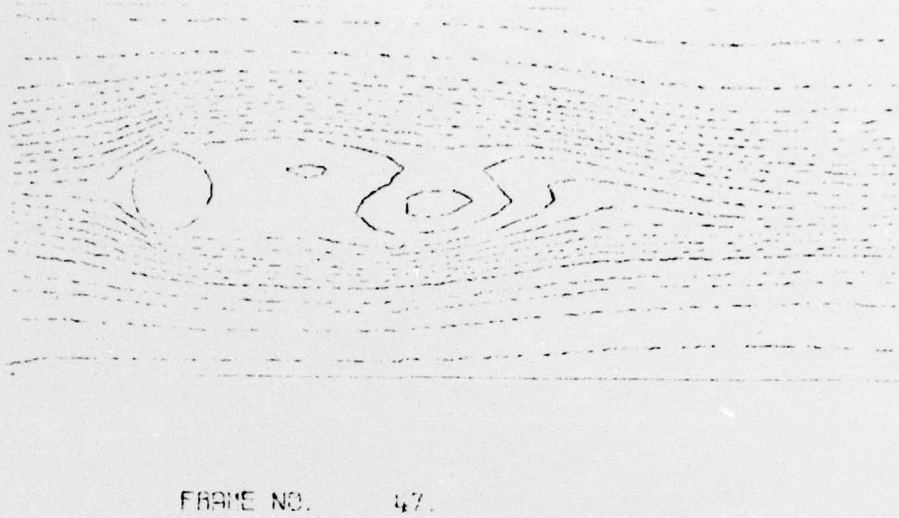


Fig. 71: Stream Function Plot at a Particular Frame for a Spherical Obstacle.

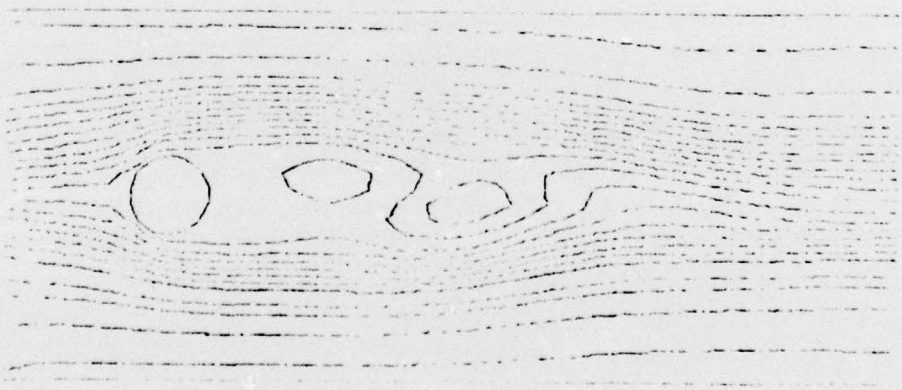


Fig. 72: Stream Function Plot at a Particular Time Frame for a Spherical Obstacle.

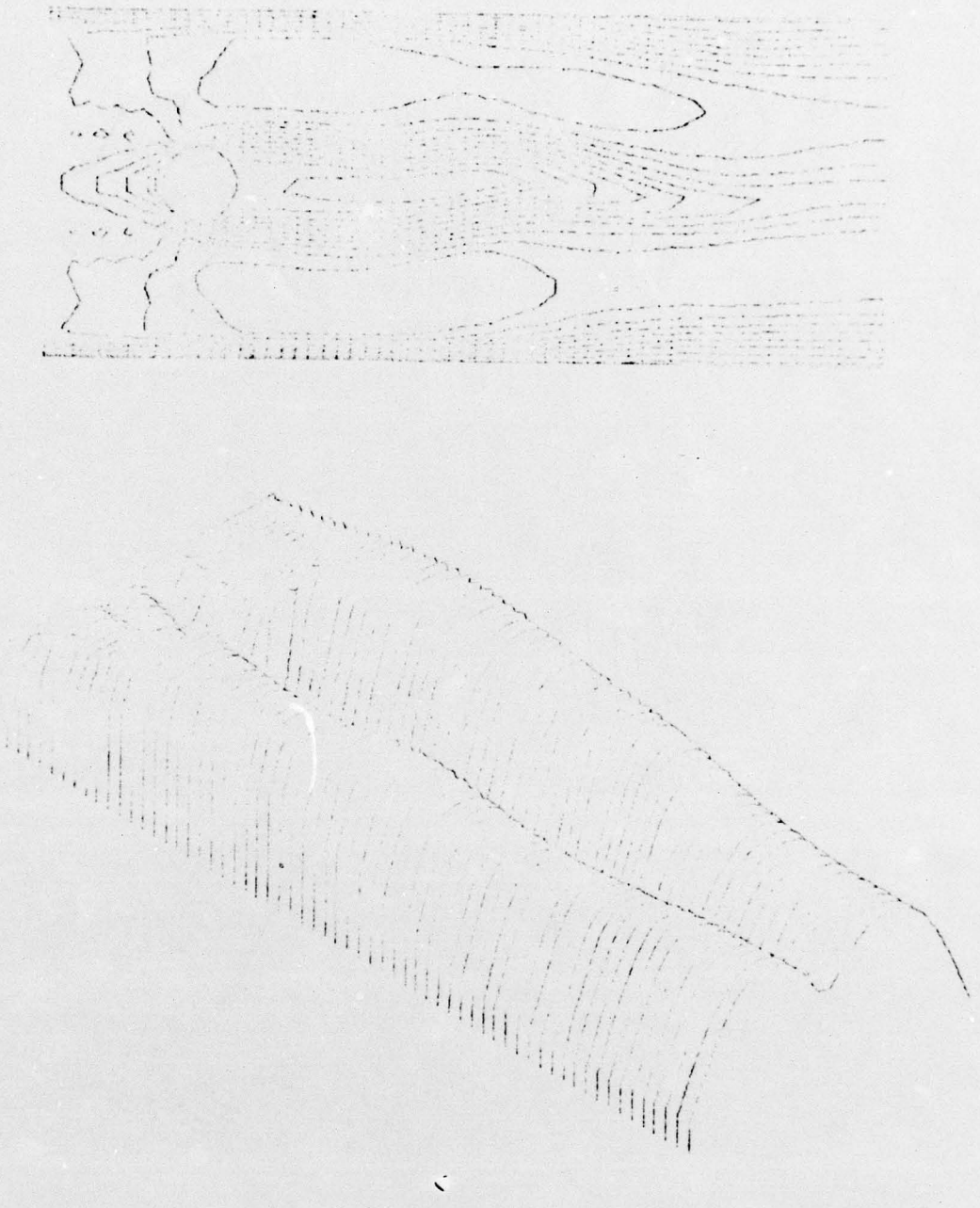


Fig. 73: The Horizontal Velocity Function Plot for a Spherical Obstacle (Surface and Isometric Views at One Time Frame)

Fig. 74: The Vertical Velocity Function Plot for a Spherical Obstacle (Surface and Isometric Views at One Time Frame)

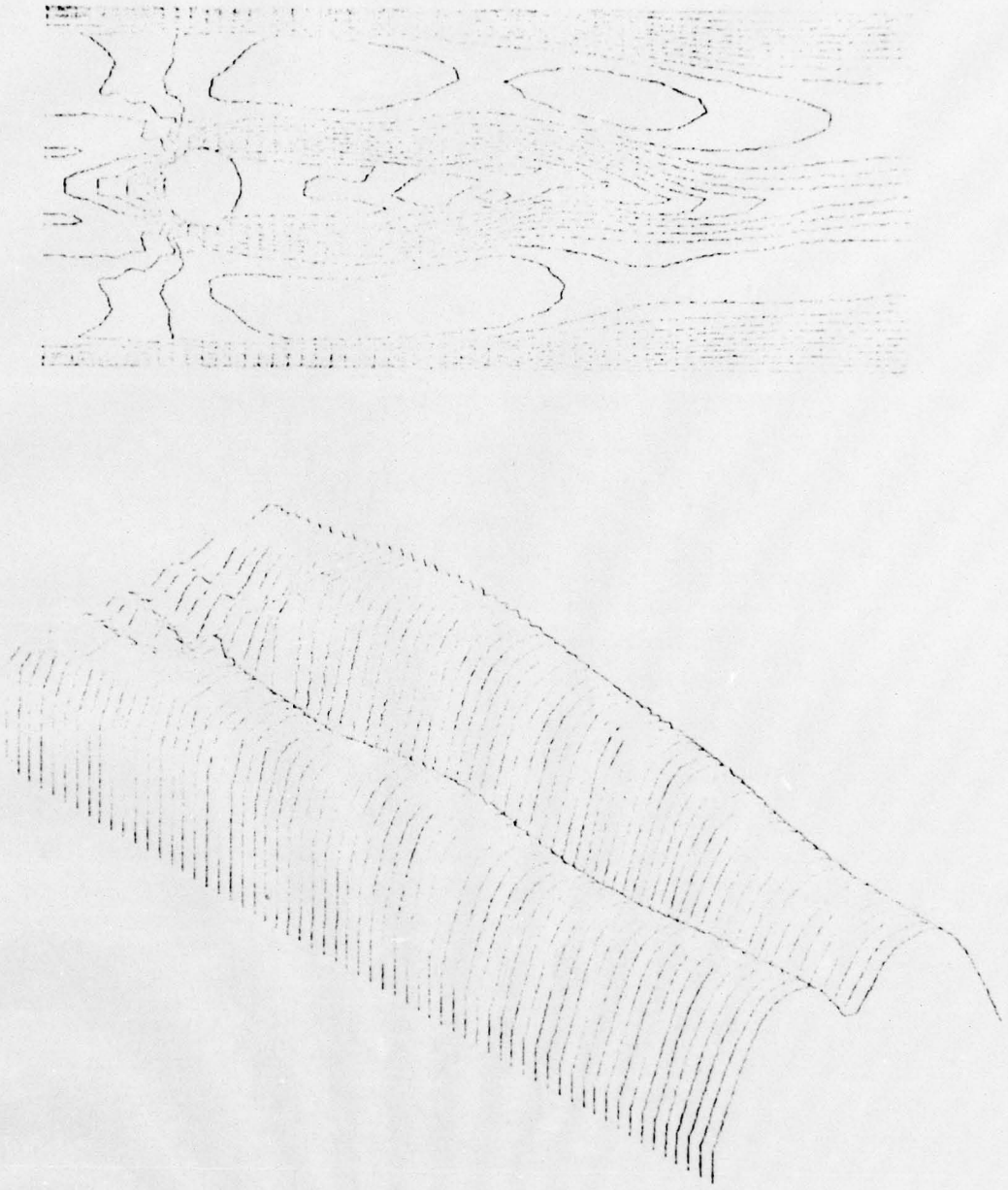


Fig. 75: The Total Velocity Function Plot for a Spherical Obstacle (Surface and Isometric Views at One Time Frame)

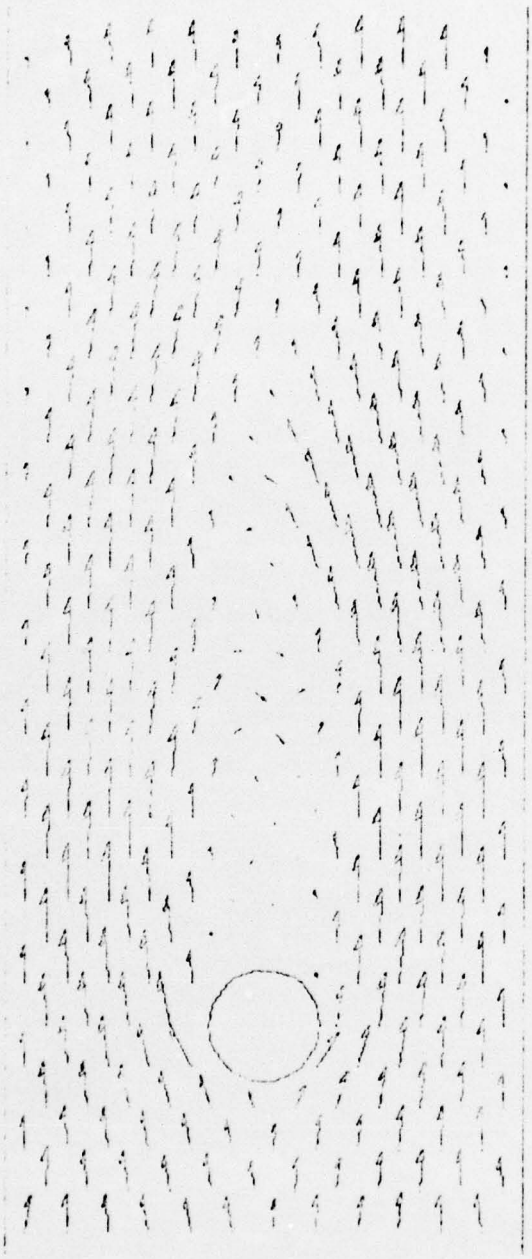


Fig. 76: The Velocity Vector Plot for a Spherical Obstacle
(Surface and Isometric Views at One Time Frame)

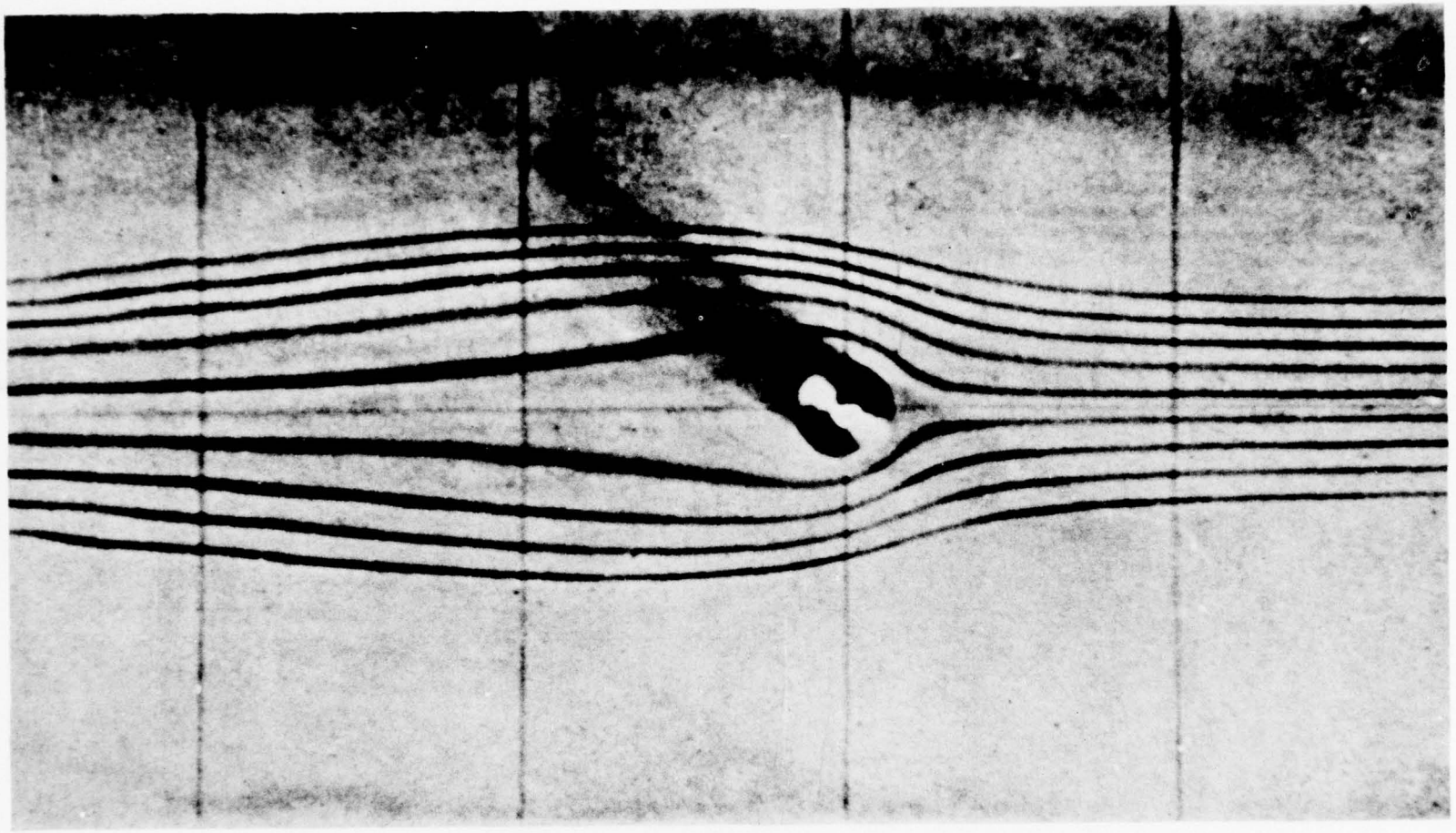


Fig. 77: An Experimental View of Flow Streamlines About a Spherical Obstacle (From Fromm[40]). Flow is From Right to Left.

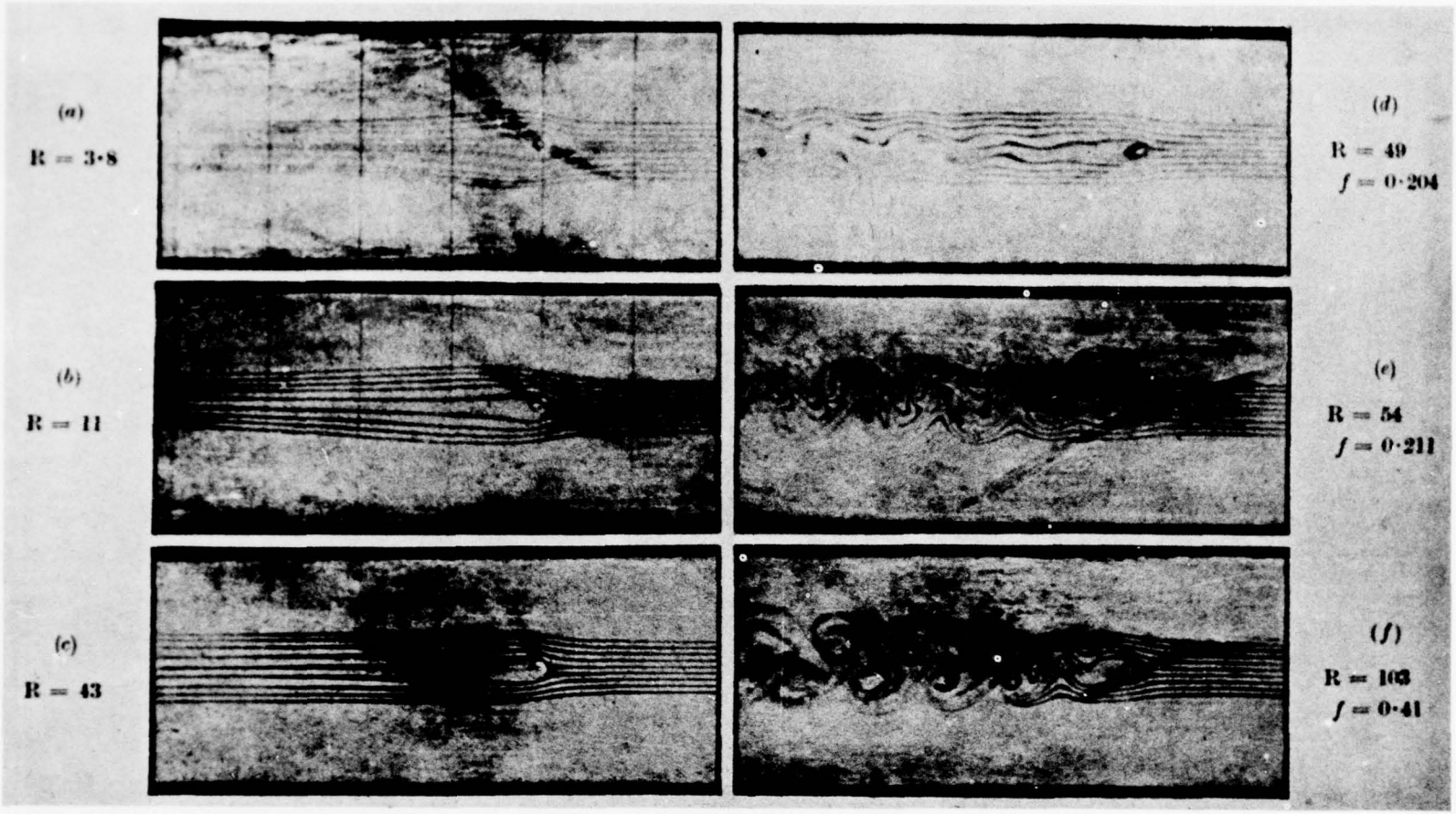
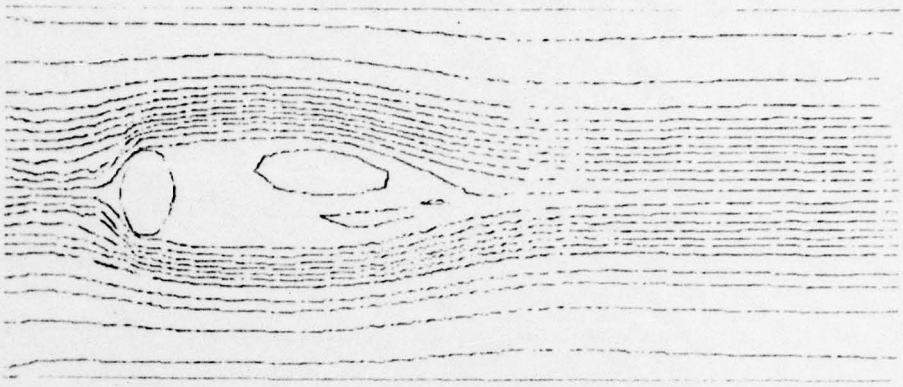


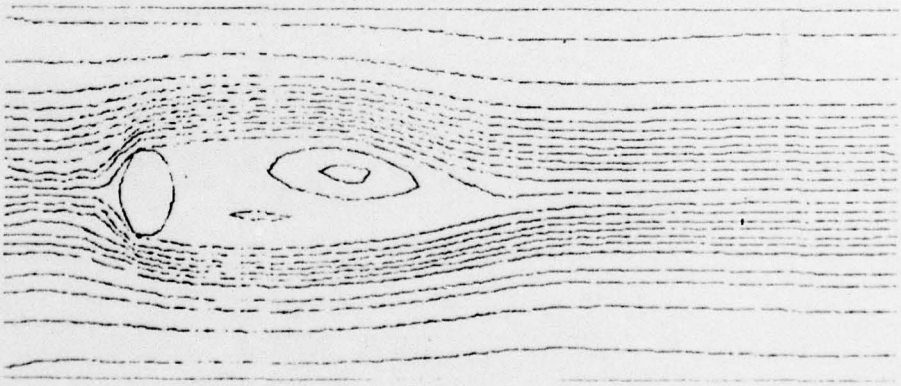
Fig. 78: Other Flow Streamlines about A Spherical Obstacle (Fromm[40])

1/2-in. cylinder in a 5-in. channel.



FRAME NO. 23.

Fig. 79: Stream Function Plot at a Particular Time Frame for An Oval Obstacle.



STREAM FUNCTION

R= 1000.

FRAME NO. 25.

Fig. 80: Stream Function Plot at a Particular Time Frame for an Oval Obstacle.

Fig. 81: Stream Function Plot at a Particular Time Frame for an Oval Obstacle.

Fig. 82: Stream Function Plot at a Particular Time Frame for an Oval Obstacle.

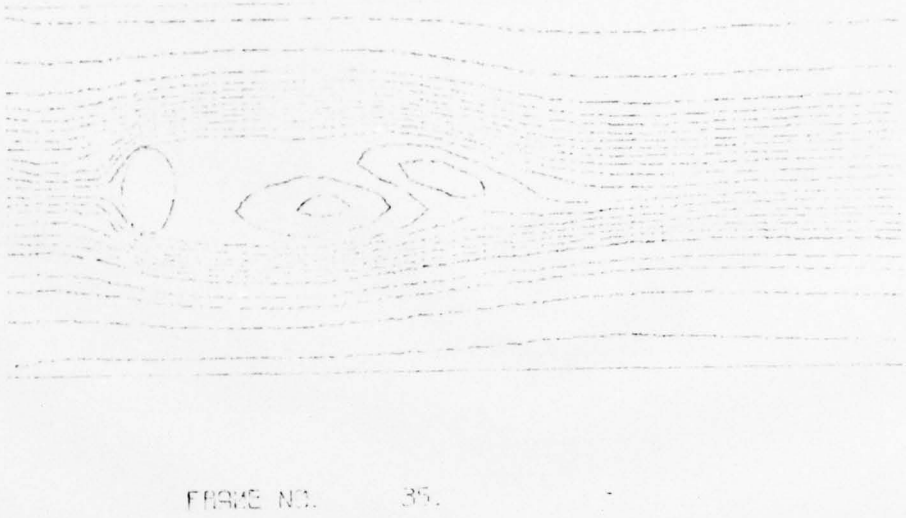


Fig. 83: Stream Function Plot at a Particular Time Frame for an Oval Obstacle.

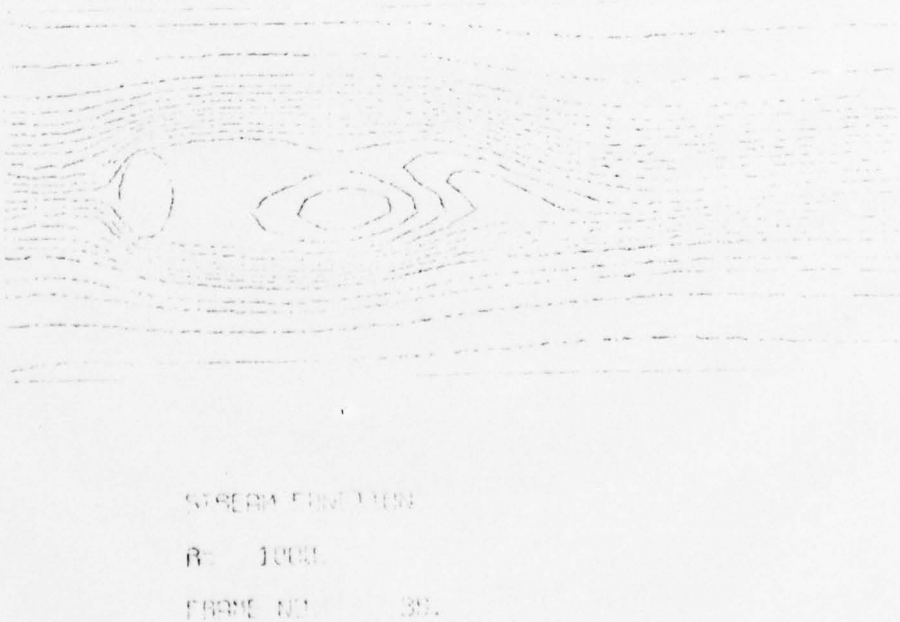


Fig. 84: Stream Function Plot at a Particular Time Frame for an Oval Obstacle.

Fig. 85: Stream Function Plot at a Particular Time Frame for an Oval Obstacle.

Fig. 86: Stream Function Plot at a Particular Time Frame for an Oval Obstacle.

Fig. 87: Stream Function Plot at a Particular Time Frame for an Oval Obstacle.

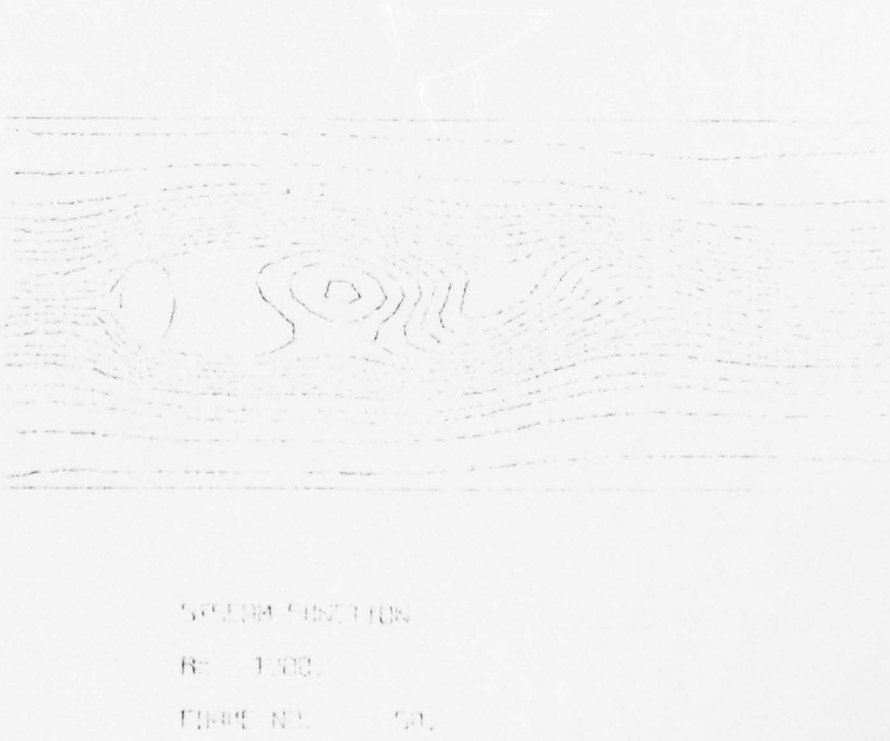


Fig. 88: Stream Function Plot at a Particular Time Frame for an Oval Obstacle.

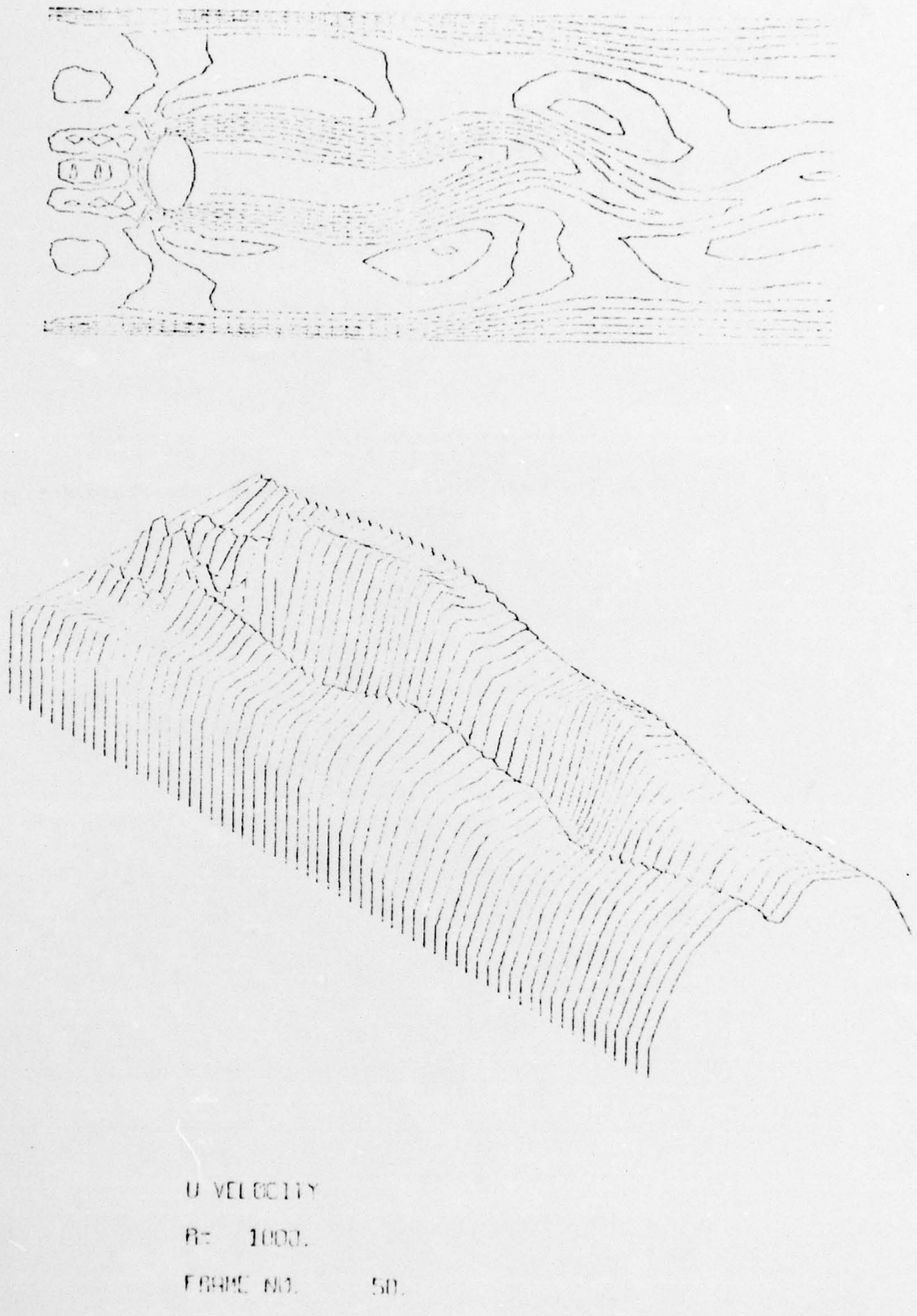


Fig. 89: The Horizontal Velocity Function Plot for an Oval Obstacle (Surface and Isometric Views at One Time Frame)

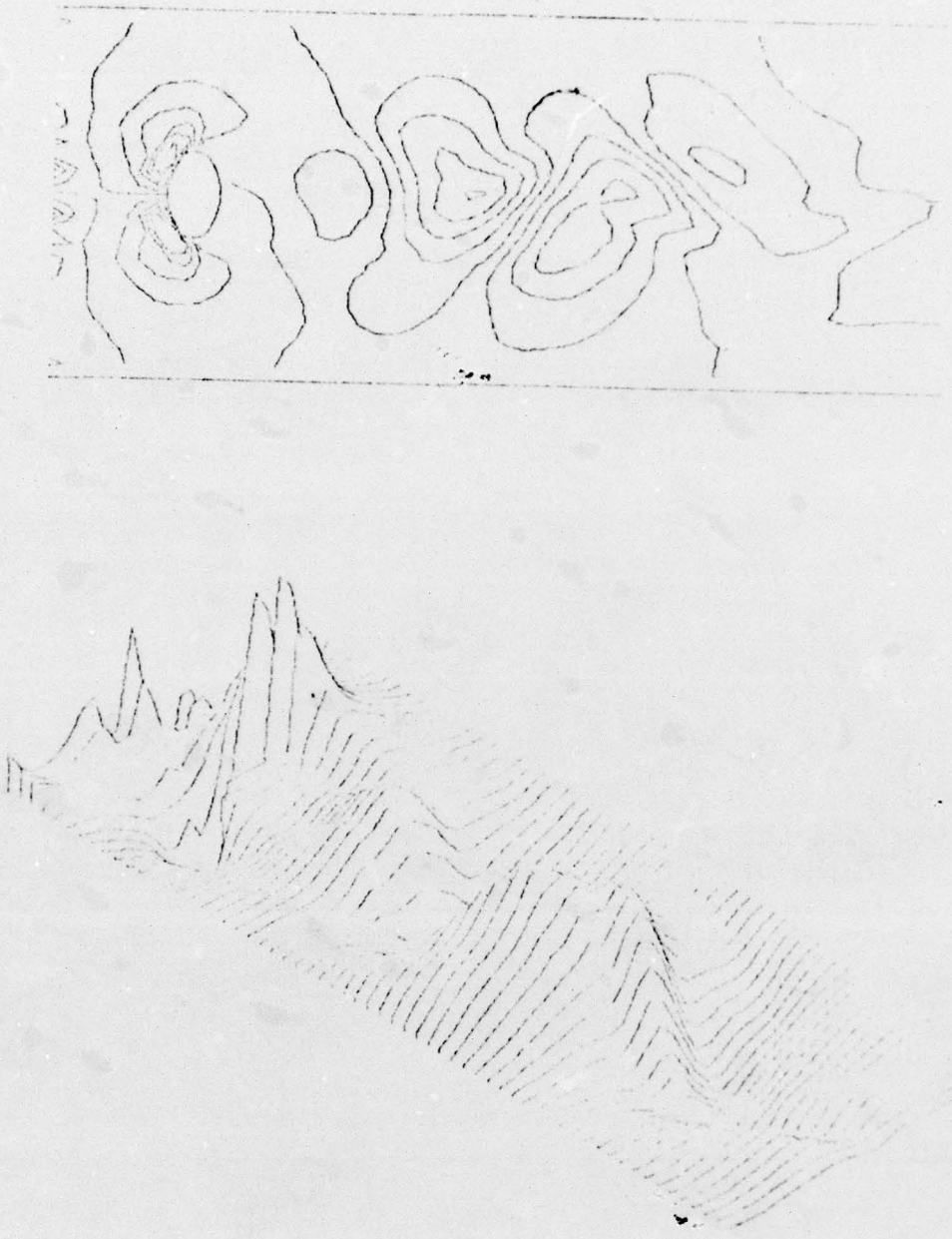


Fig. 90: The Vertical Velocity Function Plot for an Oval Obstacle (Surface and Isometric Views at One Time Frame)

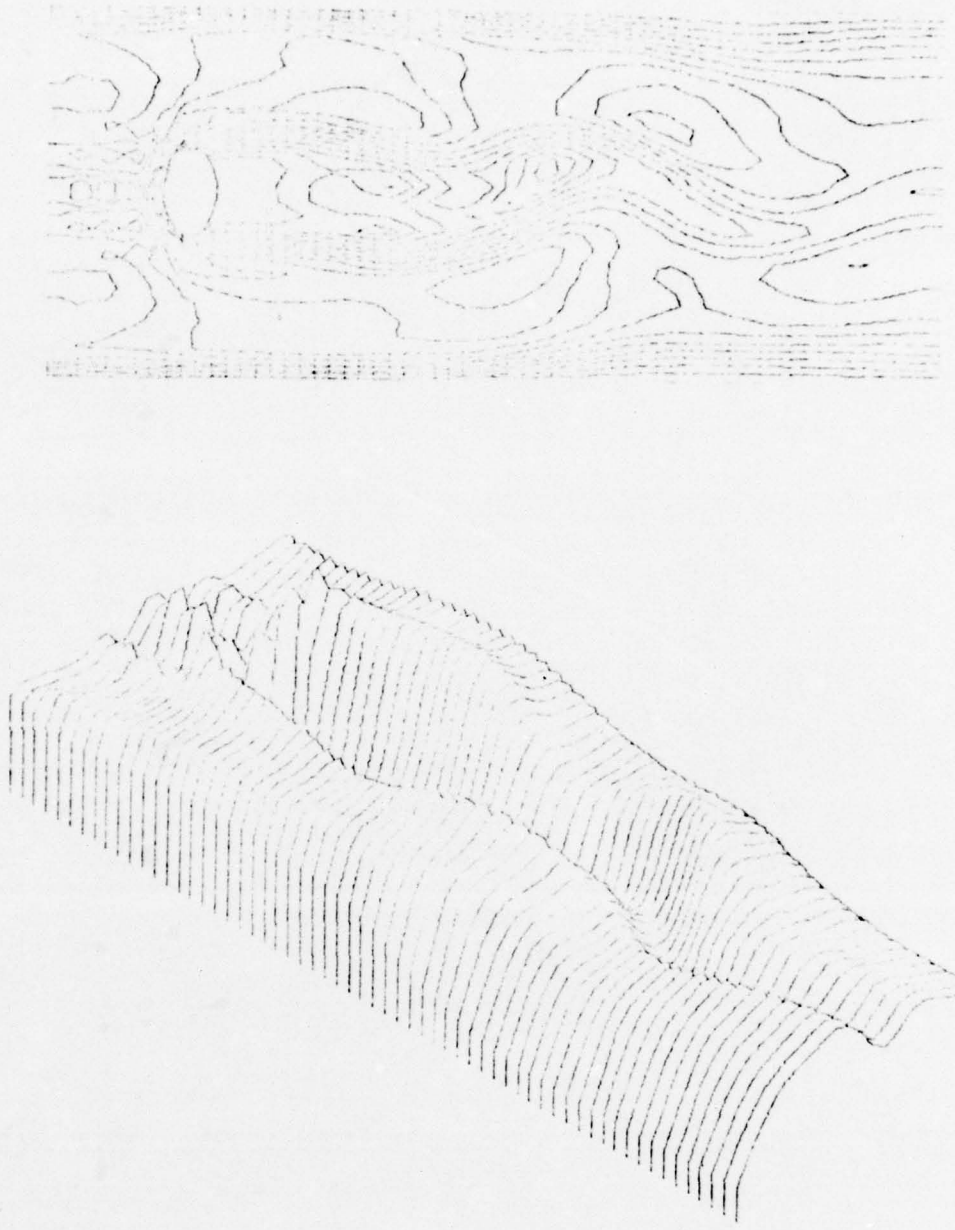


Fig. 91: The Total Velocity Function Plot for an Oval Obstacle (Surface and Isometric Views at One Time Frame)

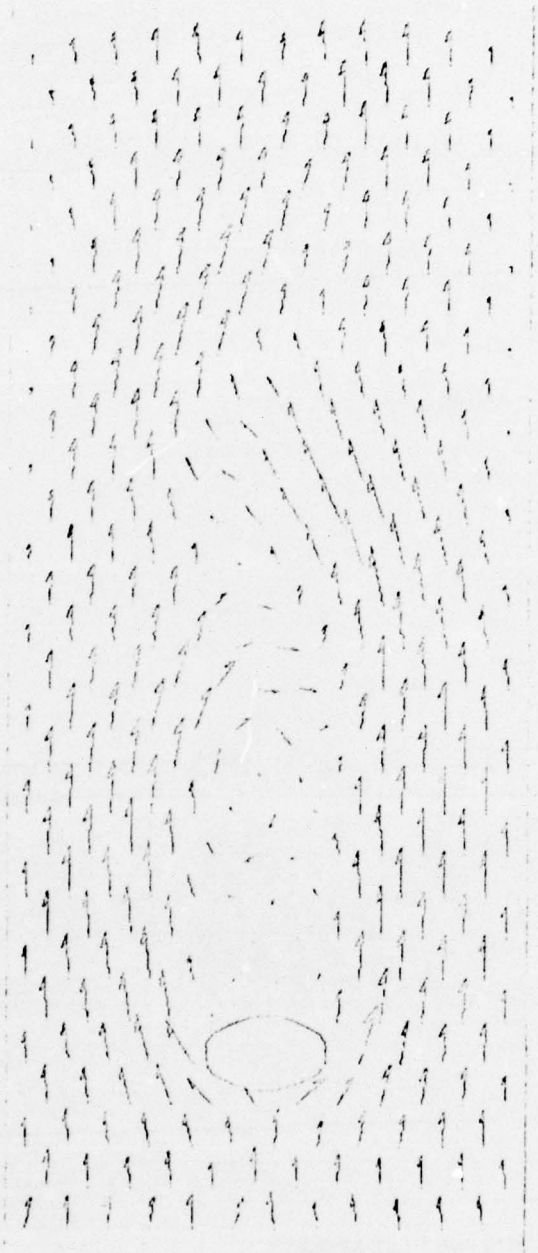
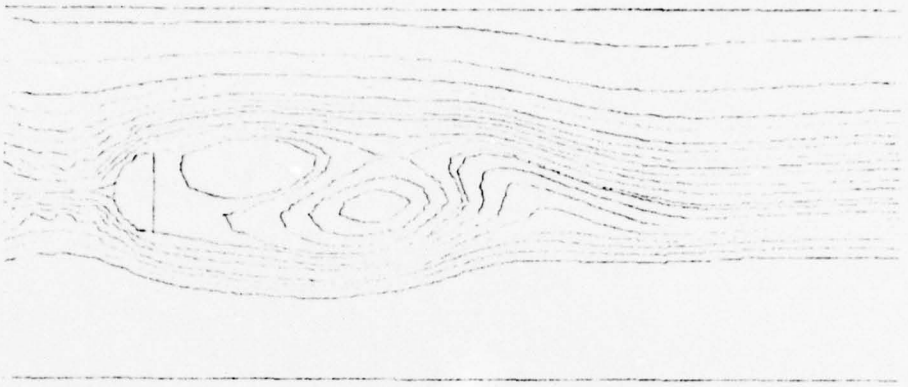
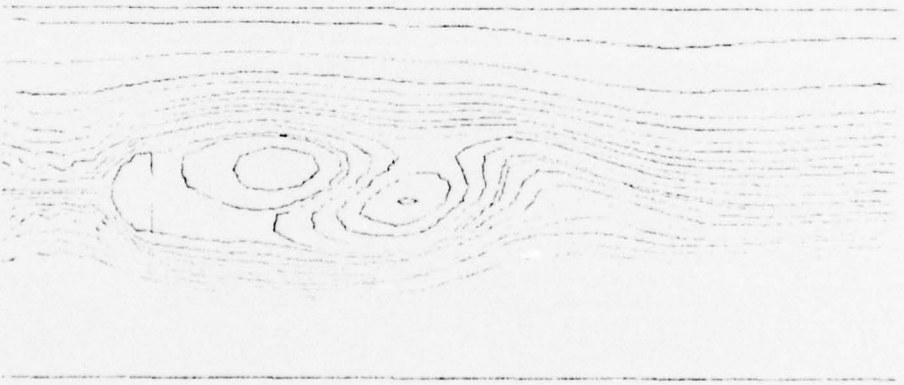


Fig. 92: The Velocity Vector Plot for an Oval Obstacle
(Surface and Isometric Views at One Time Frame)



FRAME NO. 41.

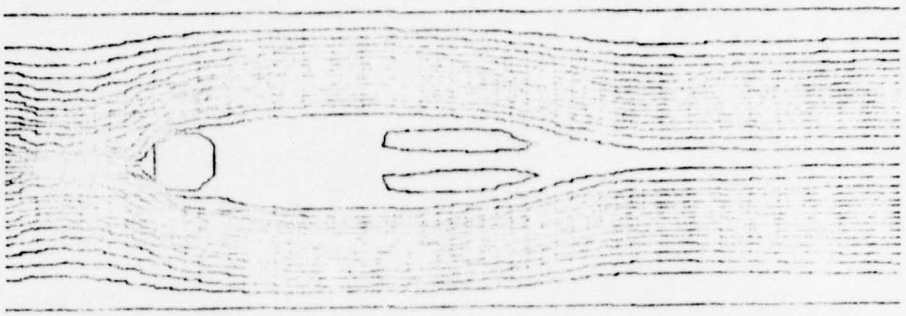


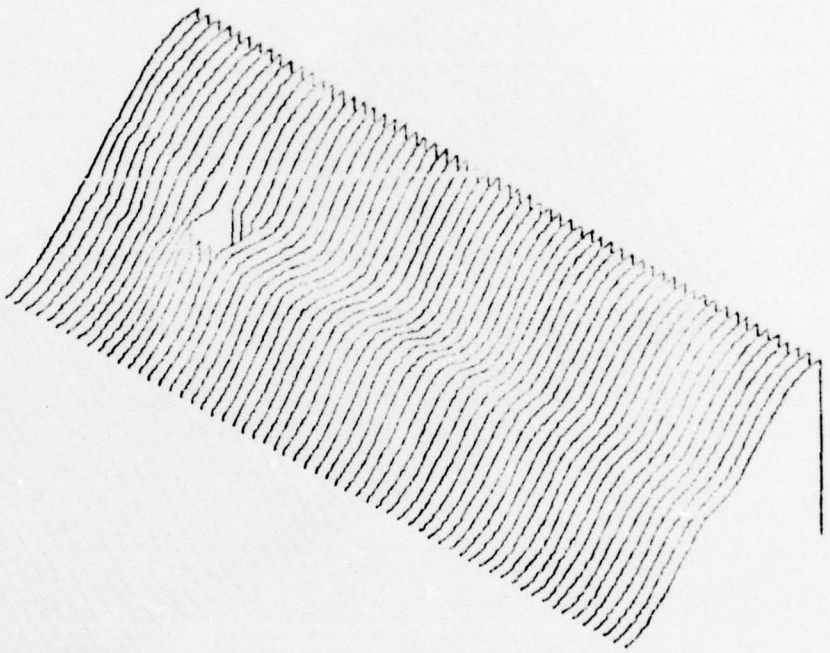
STREAM FUNCTION

R= 1000.

FRAME NO. 44.

Fig. 93: A Stream Function Plot at a Particular Time Frame and High Reynolds Number for an Oval Obstacle (Frames 41 and 44)



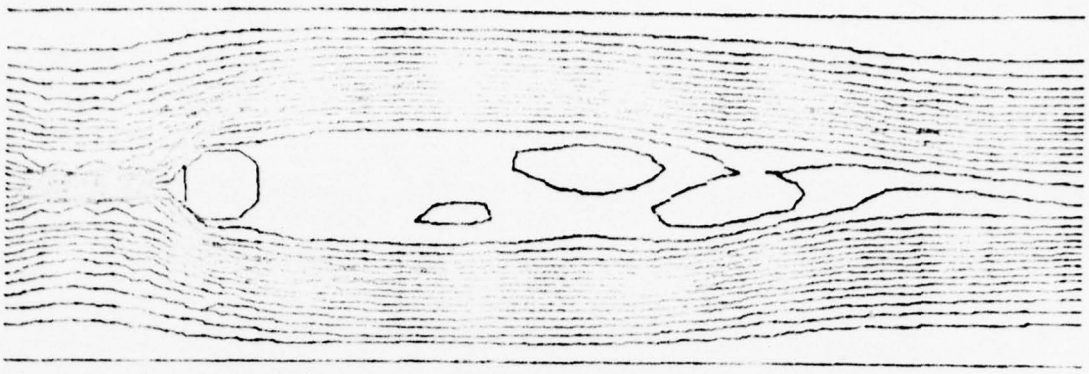


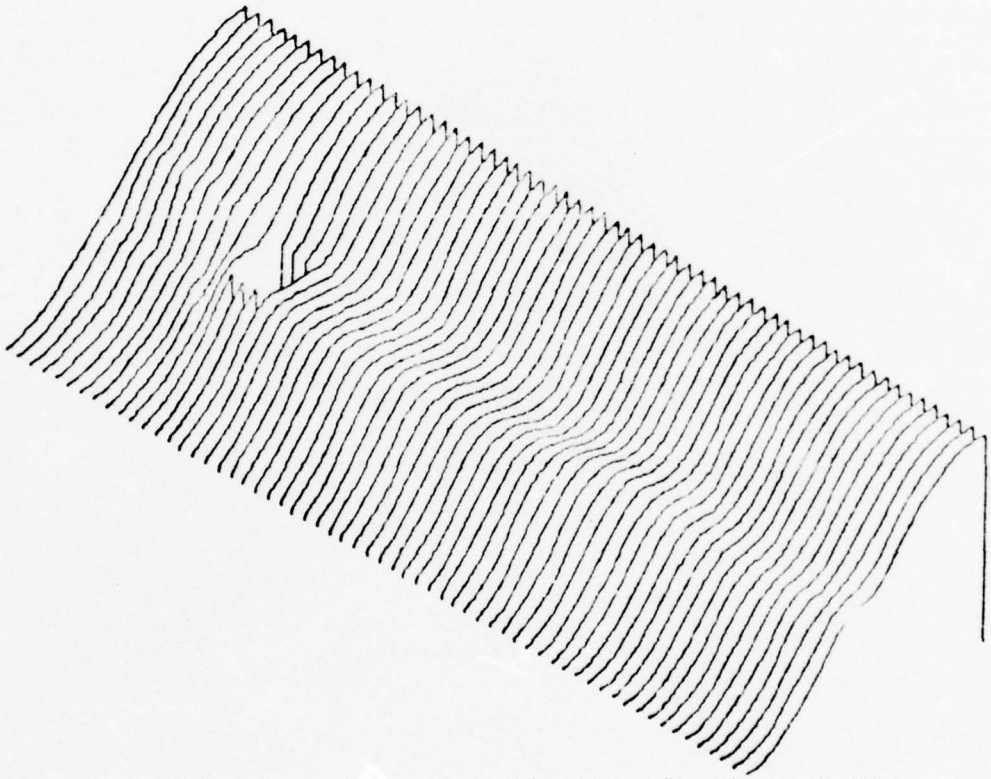
STREAM FUNCTION

R= 1000.

FRAME NO. 10.

Fig. 94: The Stream Function Plot for an Octagonal Obstacle
(Surface and Isometric Views)



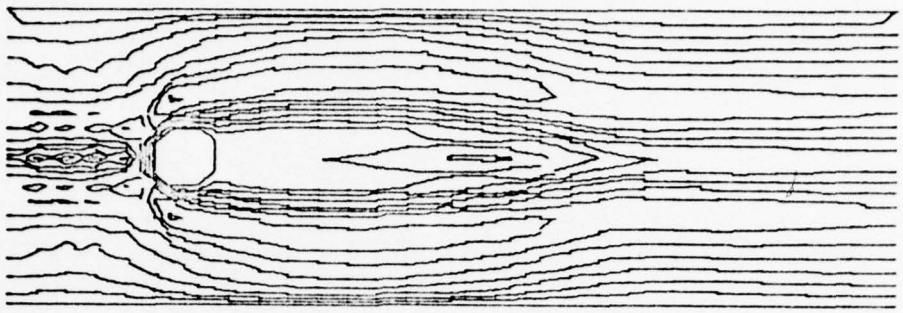


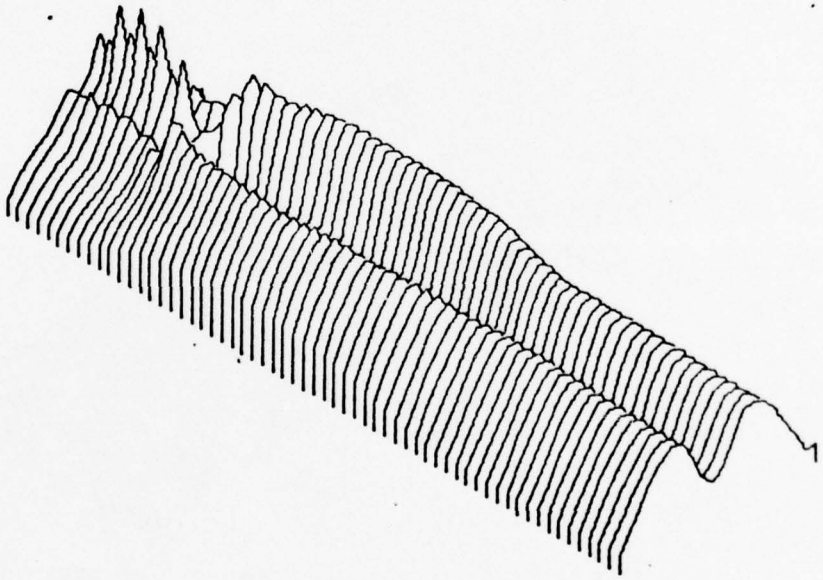
STREAM FUNCTION

R= 1000.

FRAME NO. 25.

Fig. 95: The Stream Function Plot for an Octagonal Obstacle (Surface and Isometric Views at a Later Time.



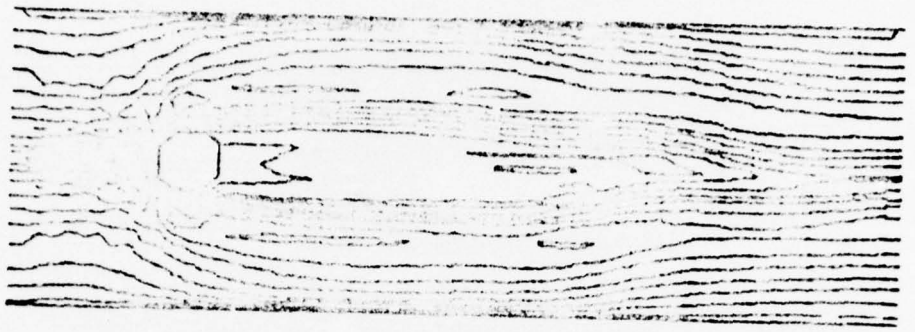


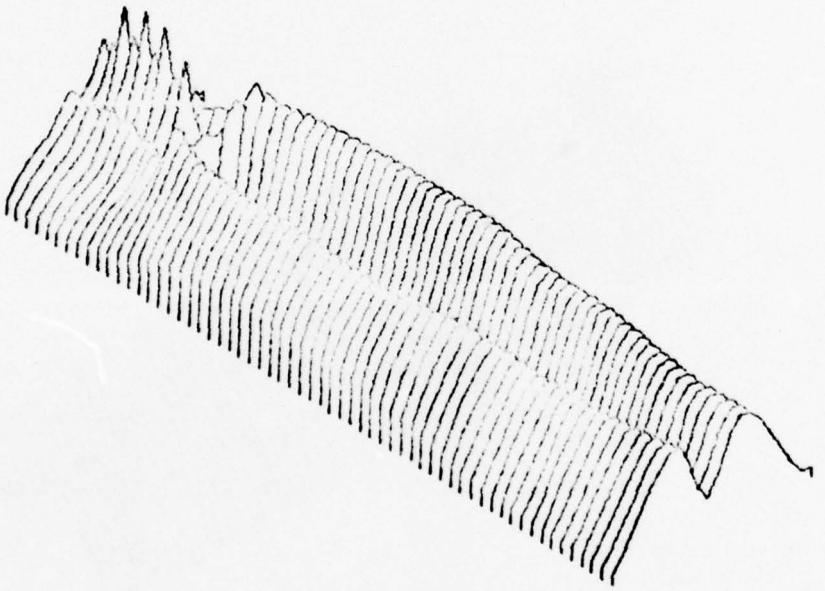
U VELOCITY

R= 1000.

FRAME NO. 10.

Fig. 96: The Horizontal Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Veiws)



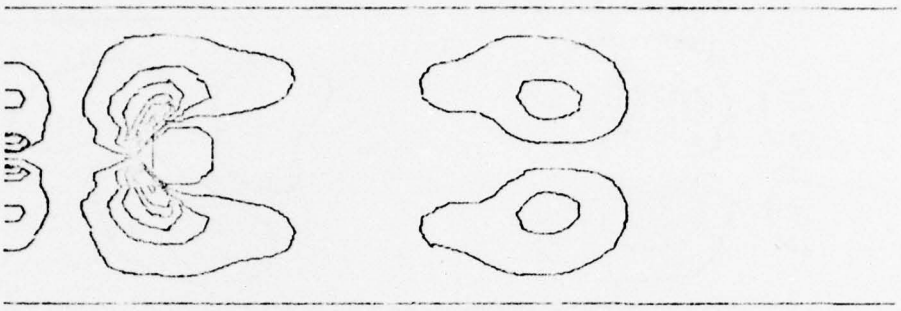


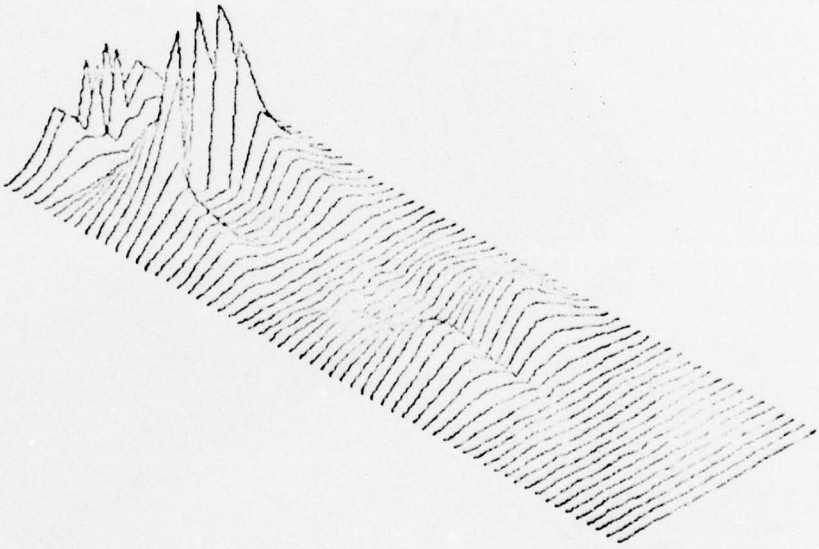
U VELOCITY

R= 1000.

FRAME NO. 25.

Fig. 97: The Horizontal Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at a Later Time)



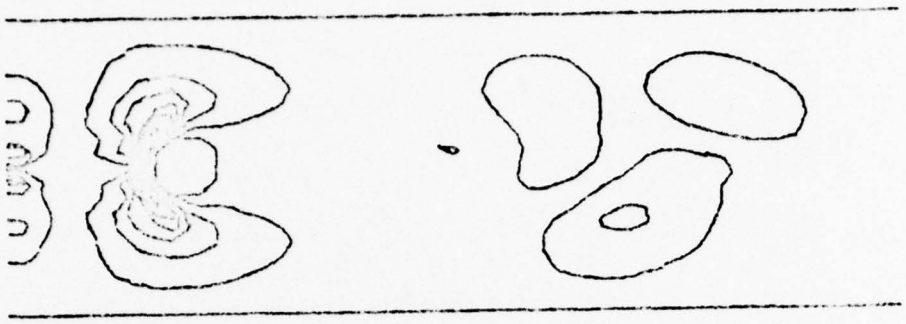


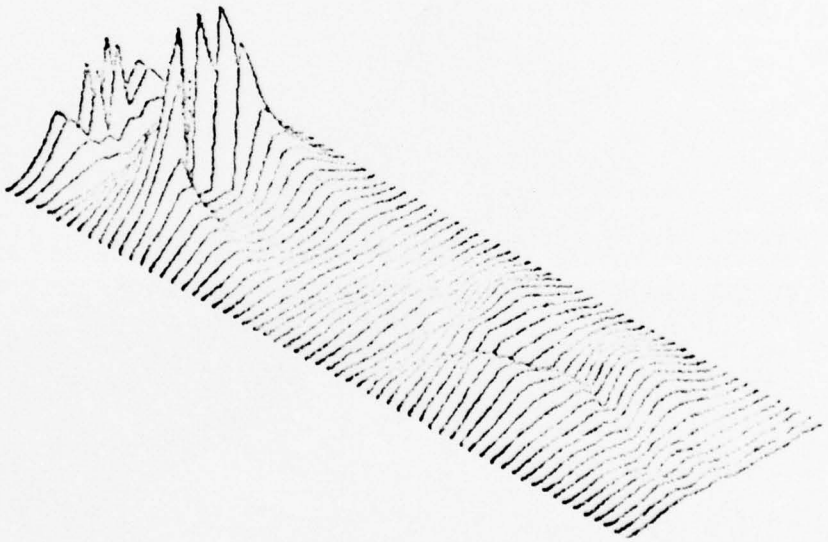
V VELOCITY

R= 1000.

FRAME NO. 10.

Fig. 98: The Vertical Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views)



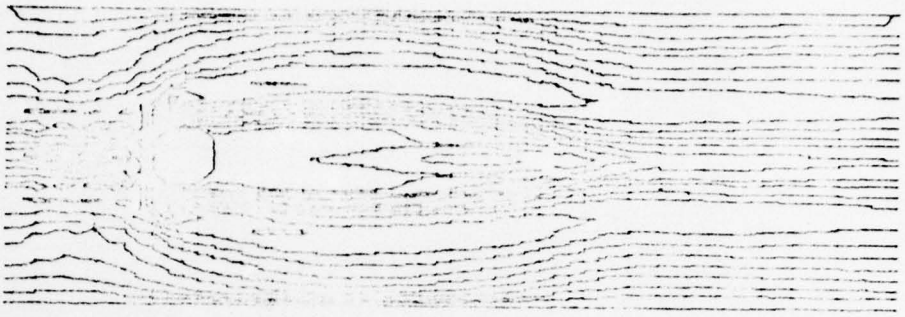


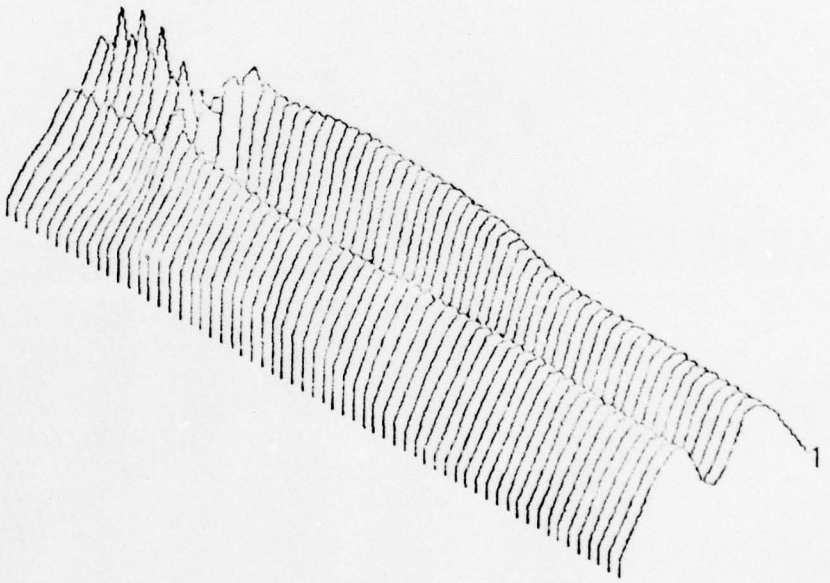
V VELOCITY

RE= 1000.

FRAME NO. 25.

Fig. 99: The Vertical Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at a Later Time)



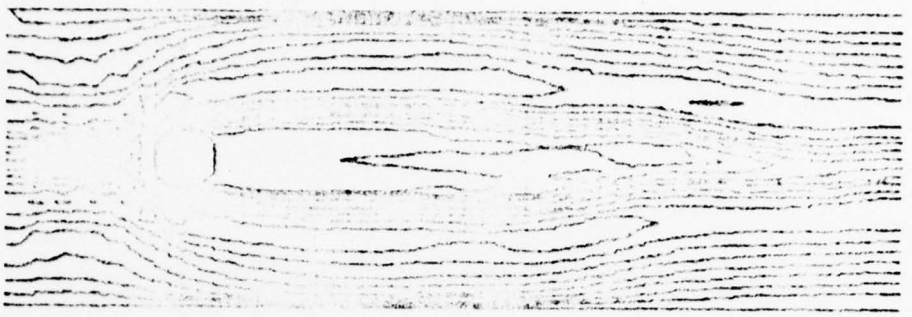


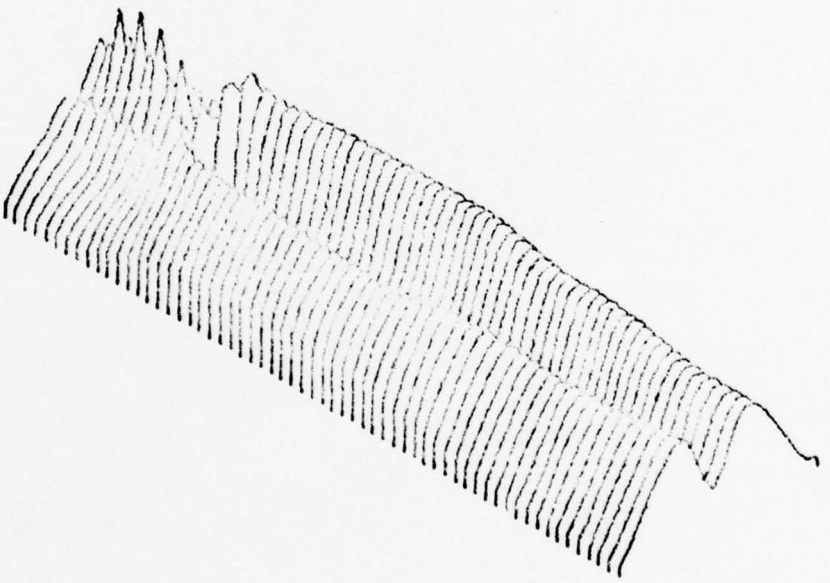
TOTAL VELOCITY

R= 1000.

FRAME NO. 10.

Fig. 100: The Total Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views)



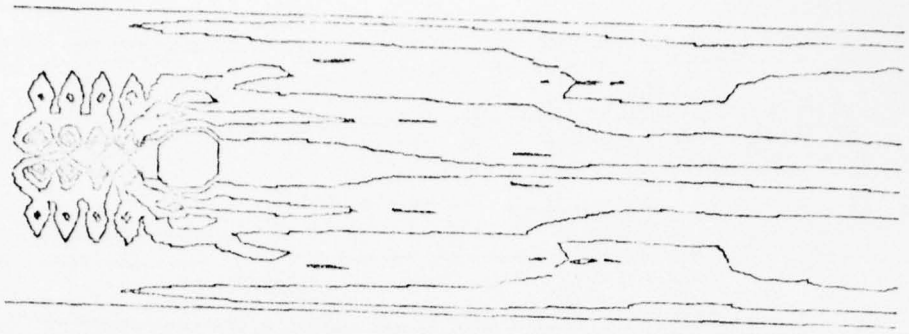


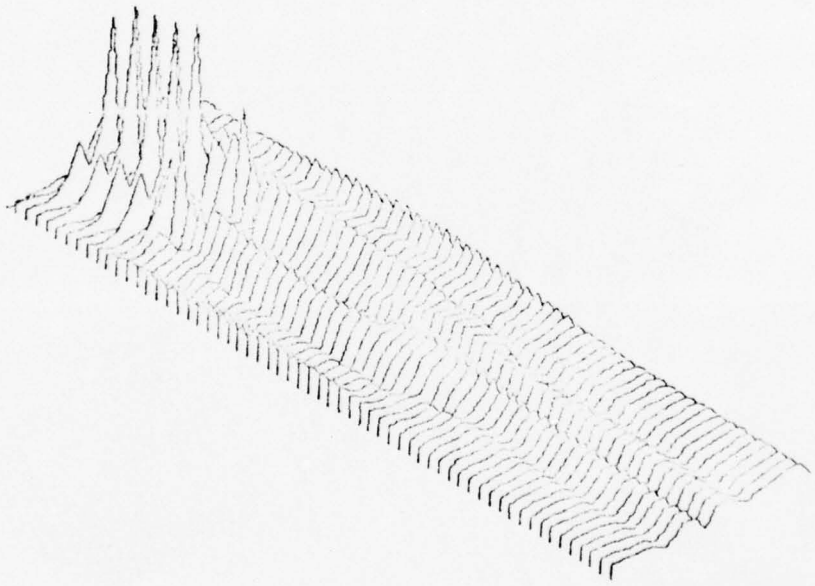
TOTAL VELOCITY

R= 1000.

FRAME NO. 25.

Fig. 101: The Total Velocity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at a Later Time)



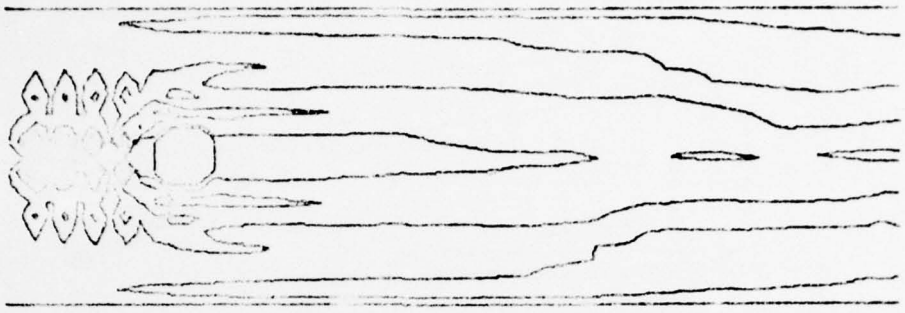


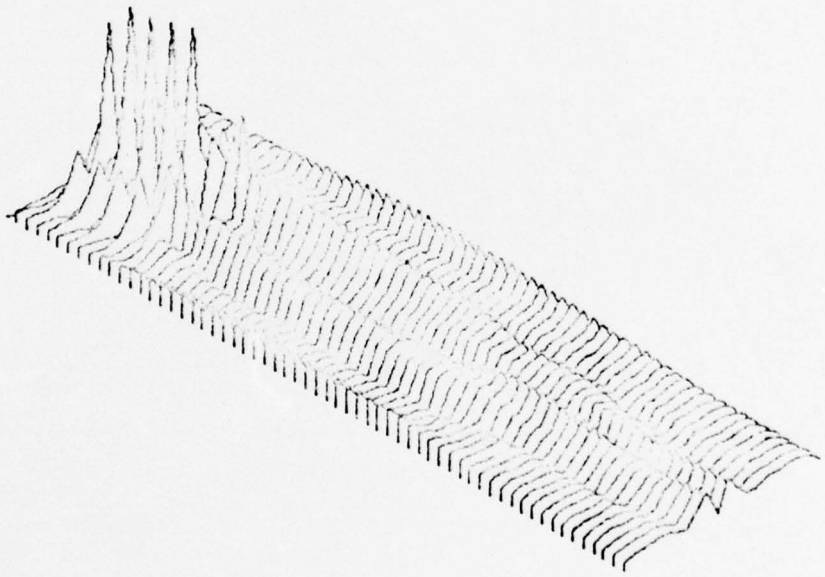
VORTICITY FUNCTION

R= 1000.

FRAME NO. 10.

Fig. 102: The Vorticity Function Plot for an Octagonal Obstacle (Surface and Isometric Views)





VORTICITY FUNCTION

RE= 1000.

FRAME NO. 25.

Fig. 103: The Vorticity Function Plot for an Octagonal Obstacle (Surface and Isometric Views at a Later Time)

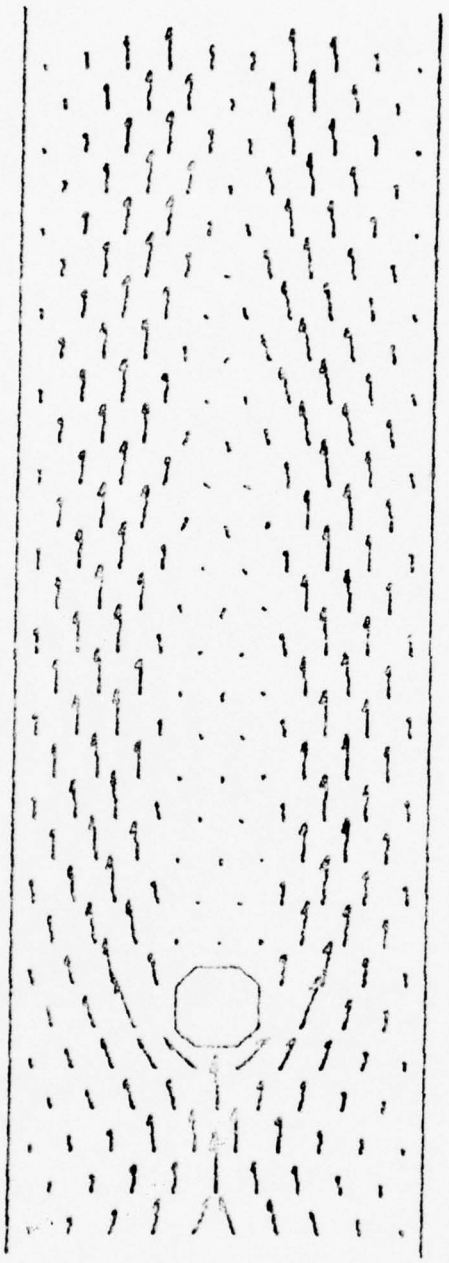


Fig. 104: The Velocity Vector Plot for an Octagonal Obstacle
(Surface and Isometric Views at a Particular Time Frame)

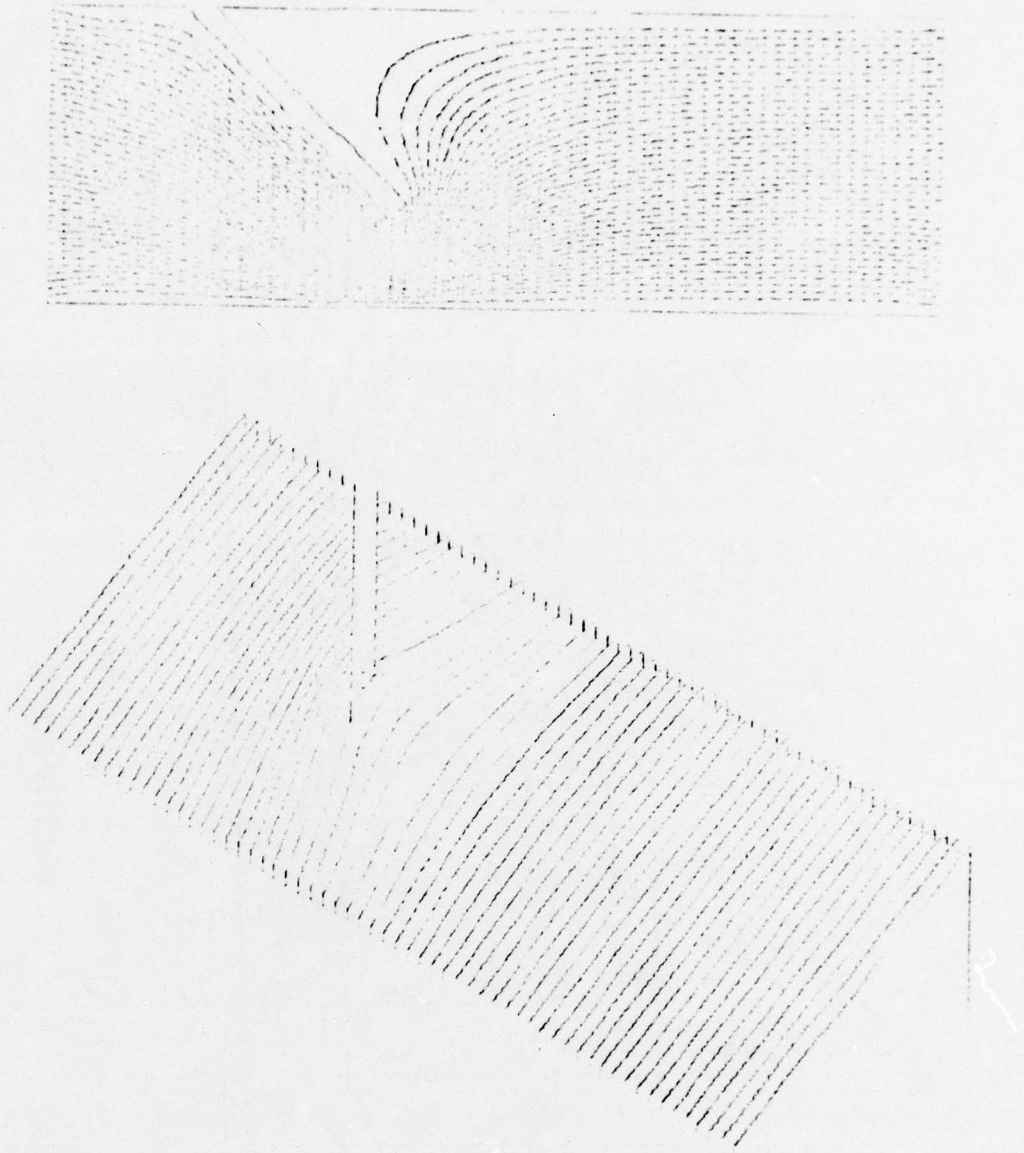


Fig. 105: Stream Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet (One Time Frame)

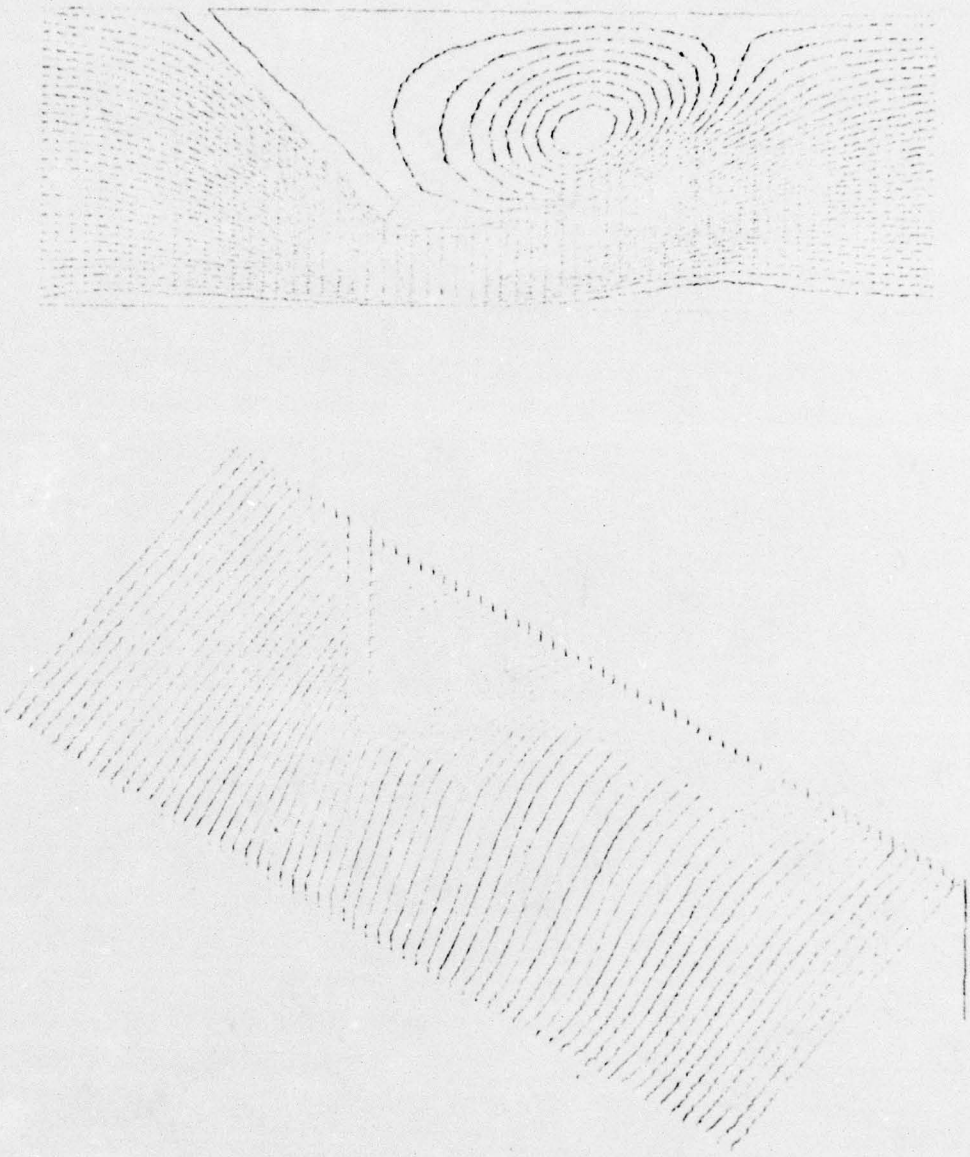


Fig. 106: Stream Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet (One Time Frame and at a Higher Reynolds Number)

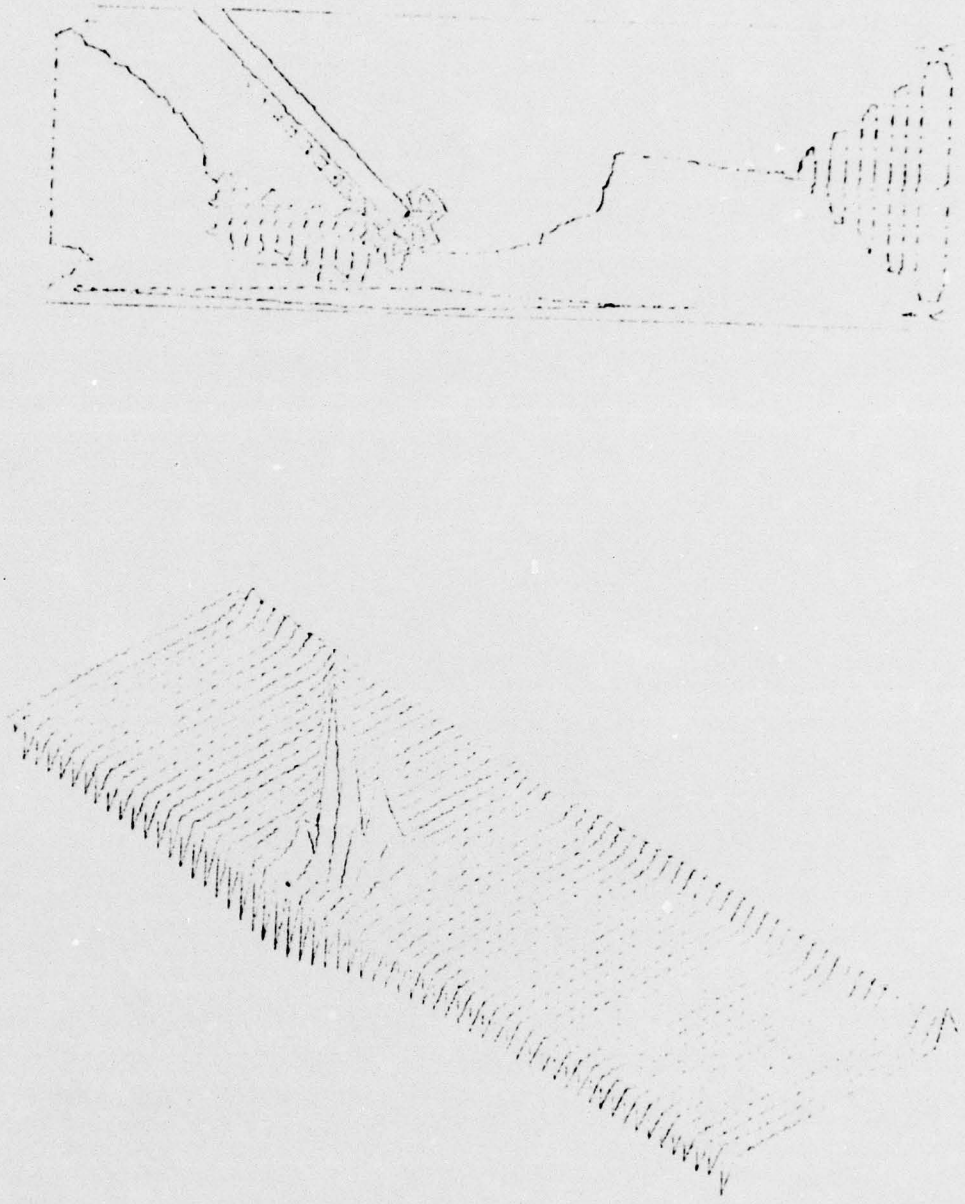


Fig. 107: Vorticity Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet (One Time Frame)

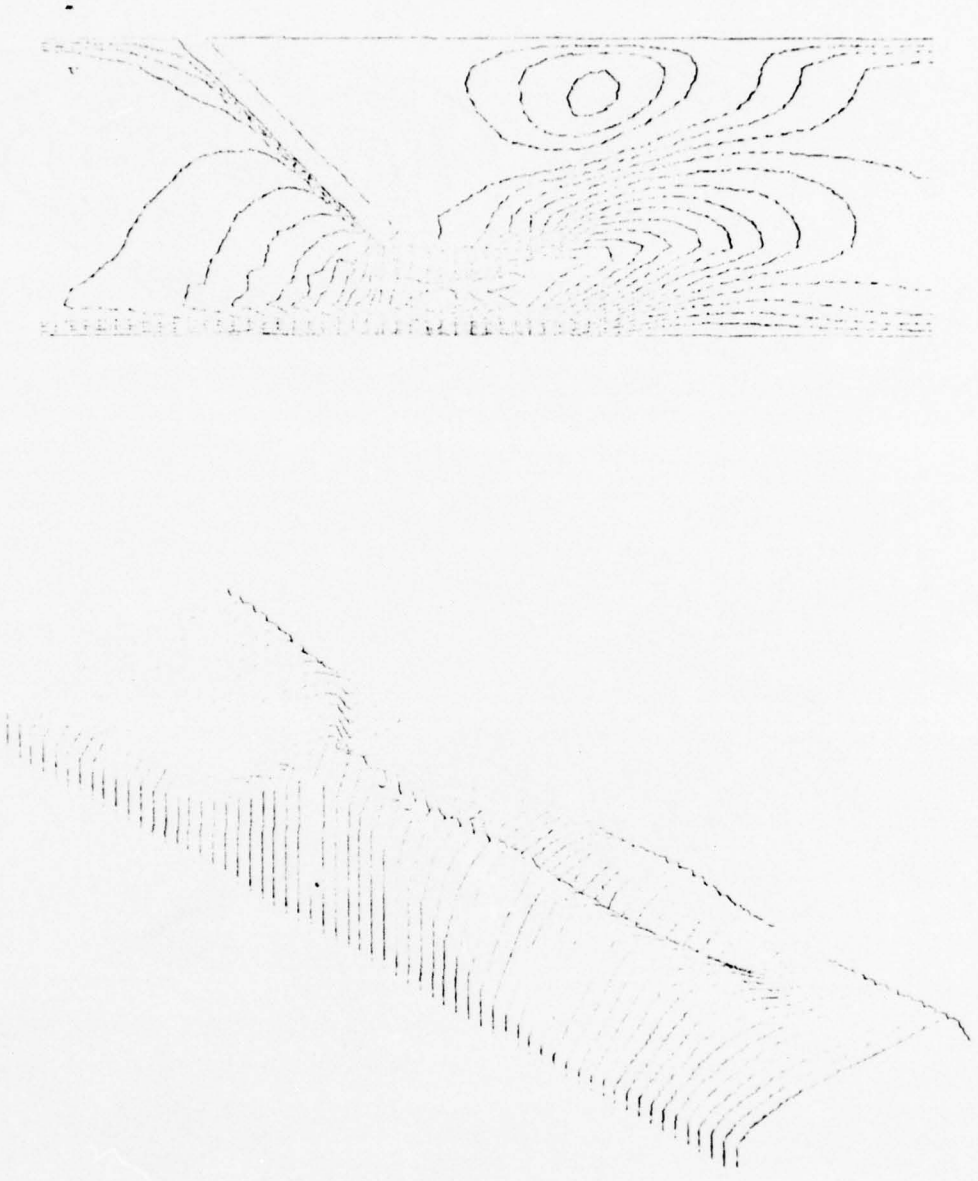


Fig. 108: Horizontal Velocity Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet (One Time Frame and at a Higher Reynolds Number)

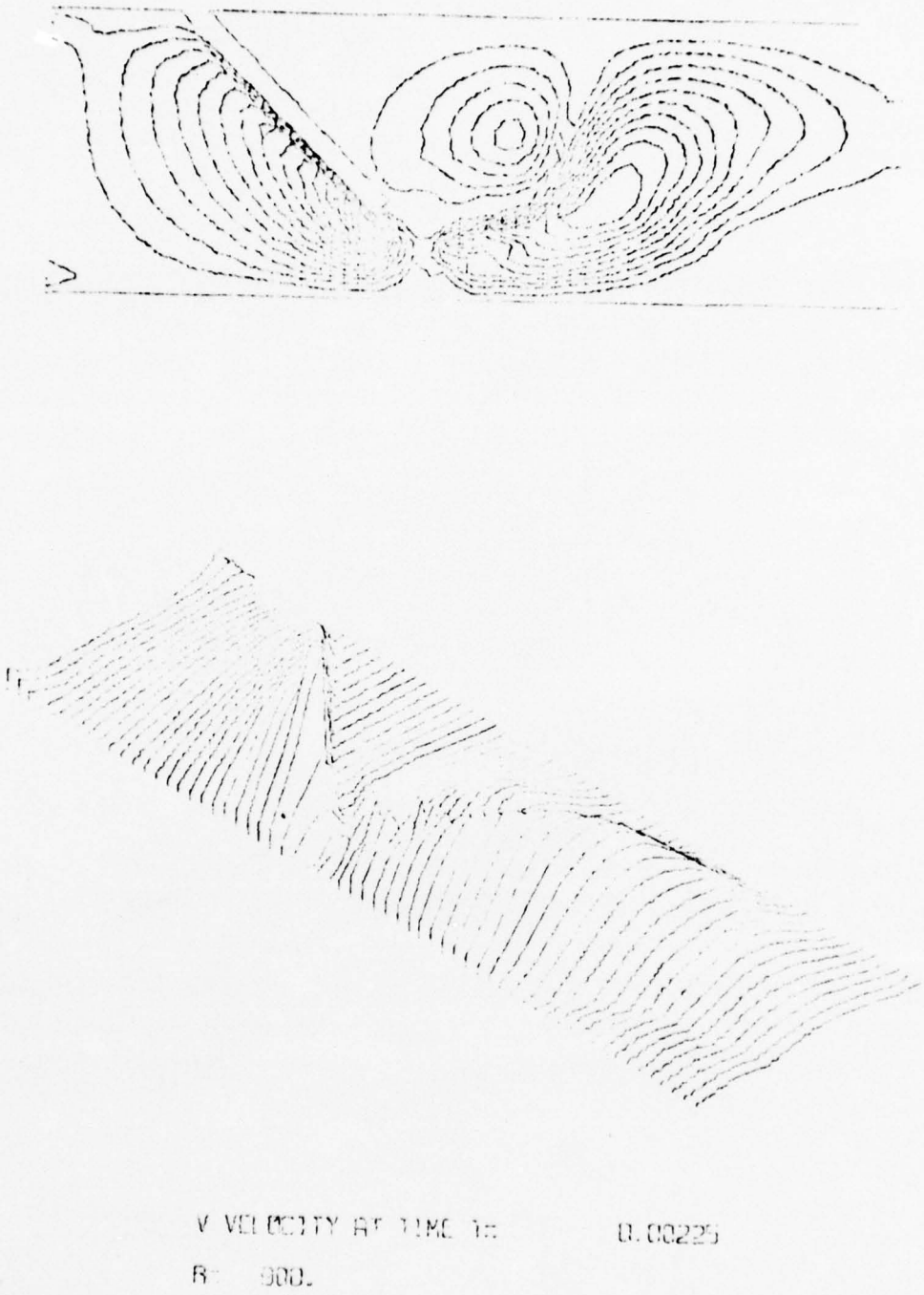


Fig. 109: Vertical Velocity Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet (One Time Frame)

Fig. 110: Total Velocity Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet (One Time Frame and at a Higher Reynolds Number)

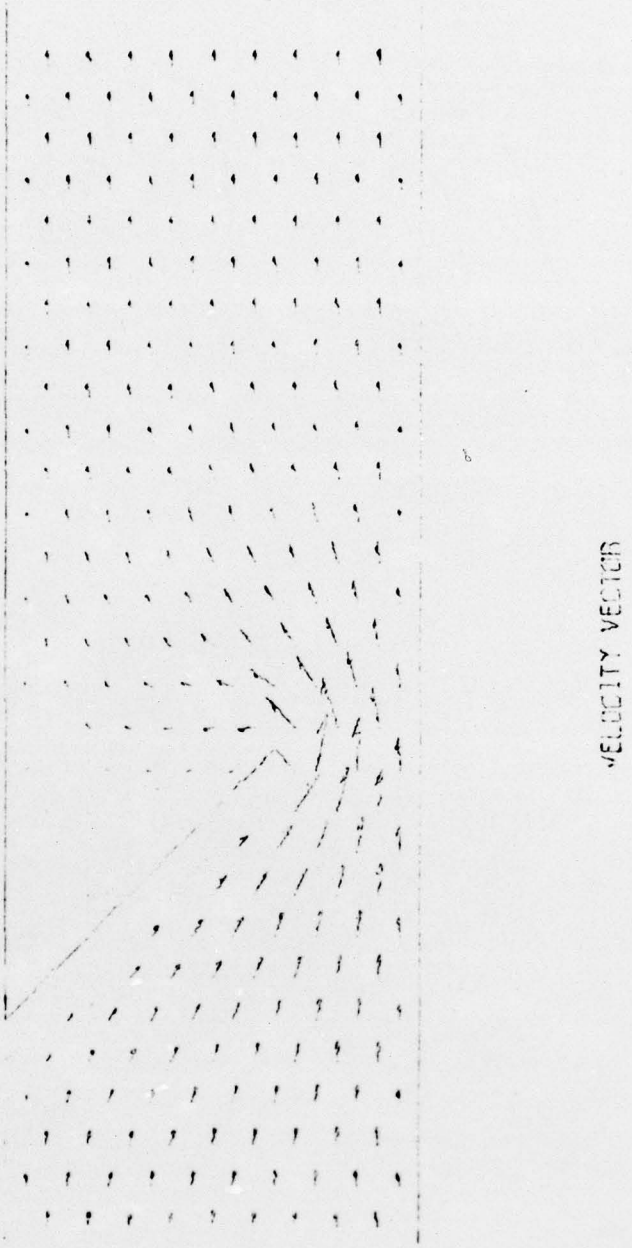


Fig. 111: Velocity Vector Function Plot for Angled One-Cusp Rigid Heart Valve Leaflet (One Time Frame)

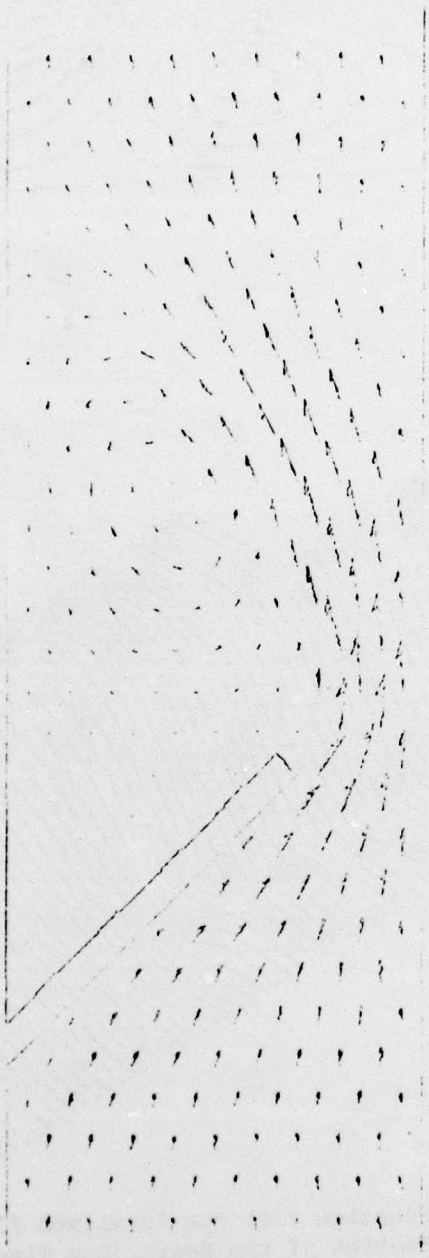


Fig. 112: Velocity Vector Plot for Angled One-Cusp Rigid Heart Valve Leaflet (One Time Frame and Higher Reynolds Number)

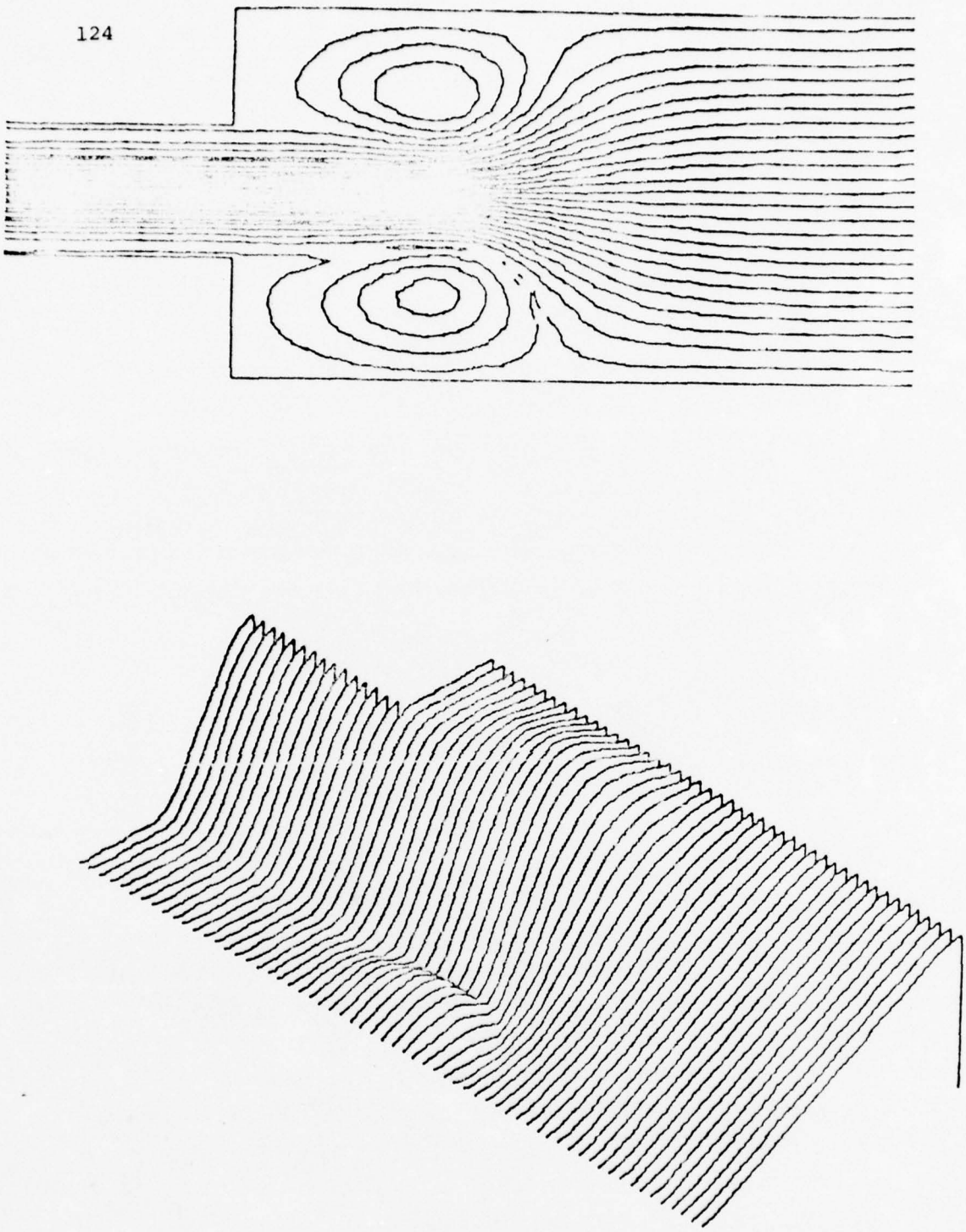
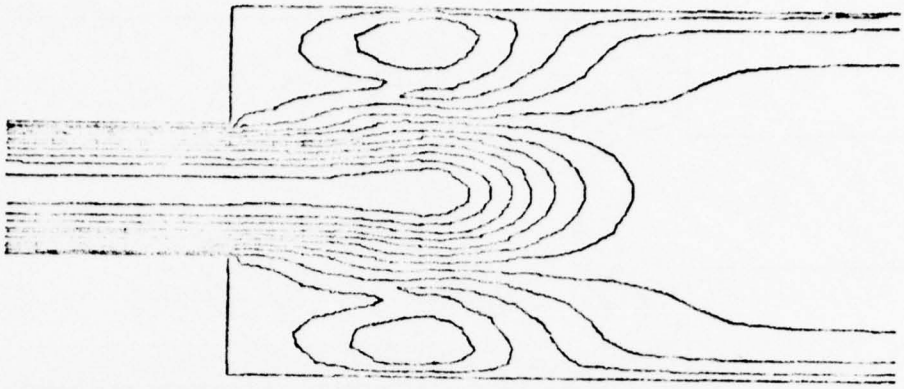
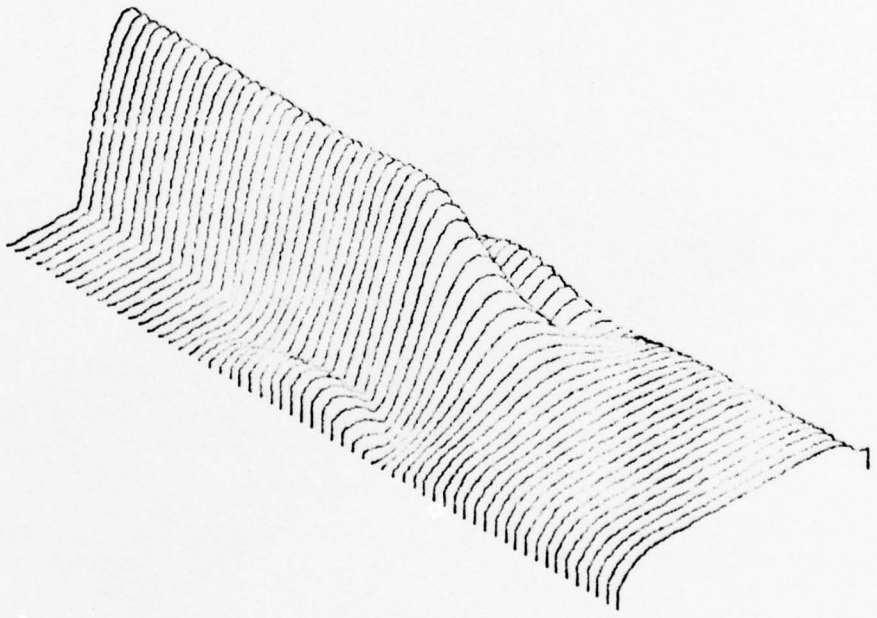


Fig. 113: Stream Function Plot for Idealized Flow into Simplified Atrium Section of the Heart (One Time Frame)



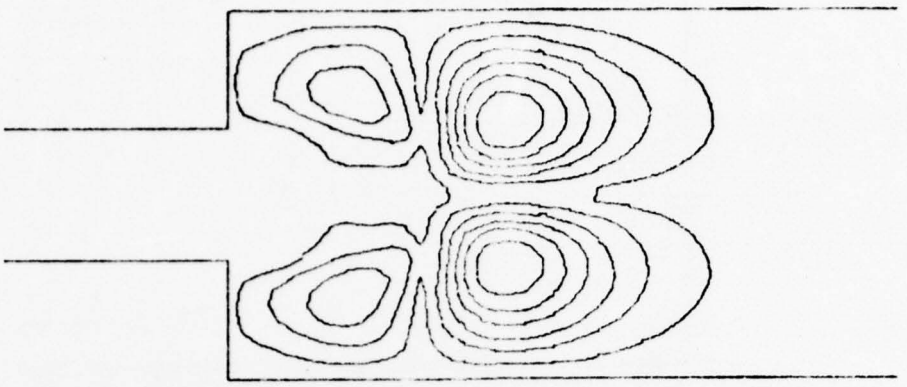


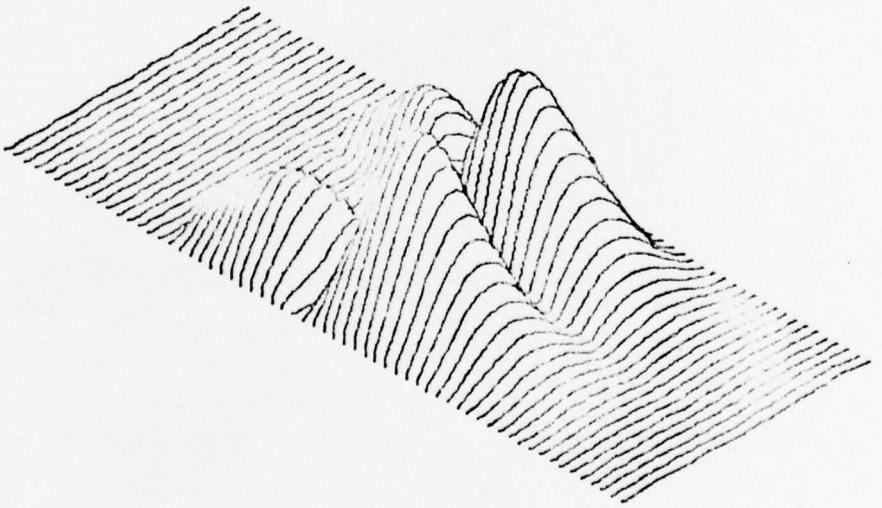
U VELOCITY

R= 300.

FRAME NO. 10.

Fig. 114: Horizontal Velocity Function Plot for Idealized Flow into Simplified Atrium Section of the Heart (One Time Frame)



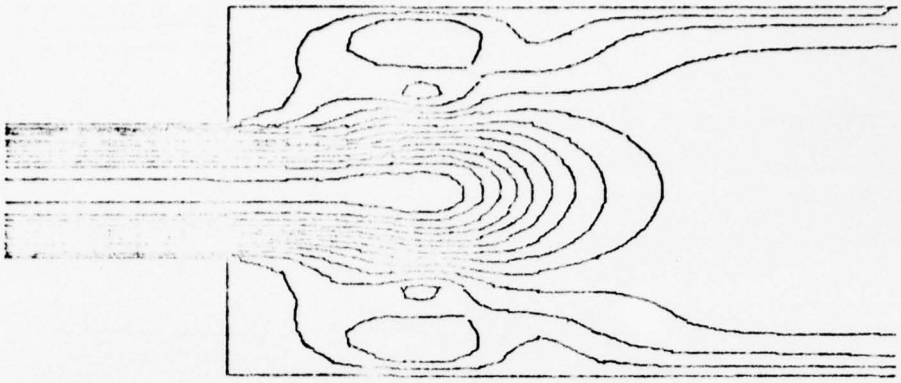


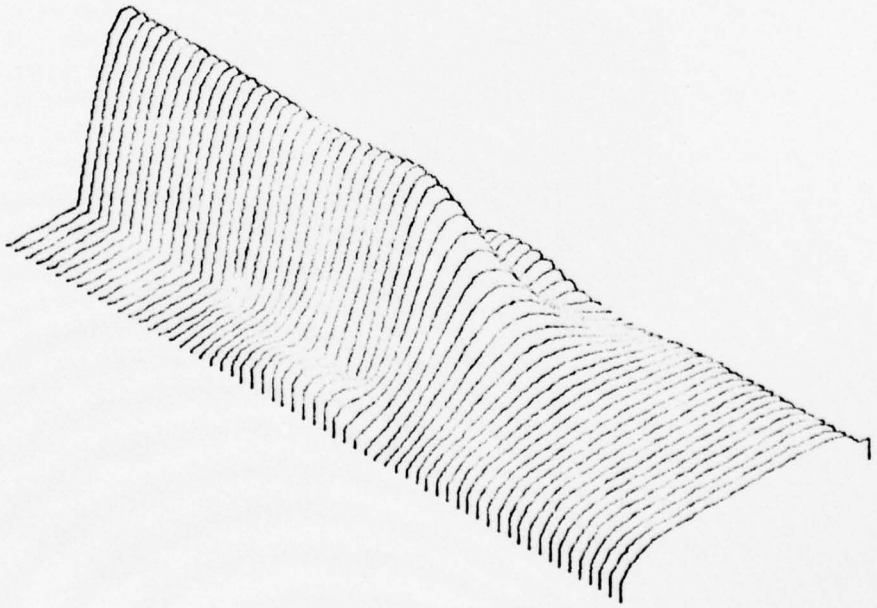
V VELOCITY

R= 300.

FRAME NO. 10.

Fig. 115: The Vertical Velocity Function Plot for Idealized Flow into Simplified Atrium Section of the Heart (One Time Frame)



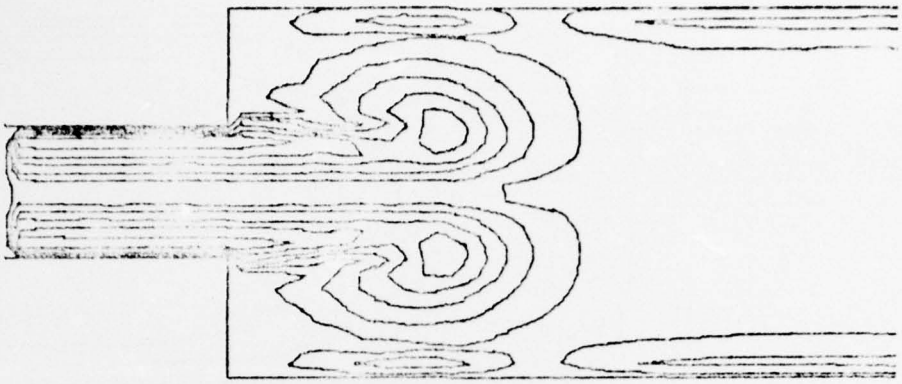


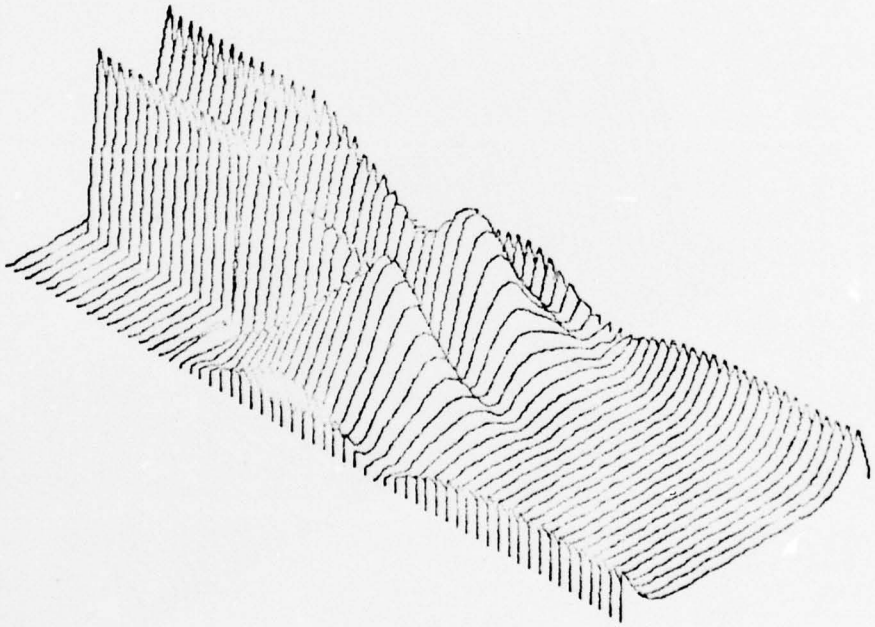
TOTAL VELOCITY

RE= 300.

FRAME NO. 10.

Fig. 116: Total Velocity Function Plot for Idealized Flow into Simplified Atrium Section of the Heart (One Time Frame)





VORTICITY FUNCTION

R= 300.

FRAME NO. 10.

Fig. 117: Vorticity Function Plot for Idealized Flow into Simplified Atrium Section of the Heart (One Time Frame)

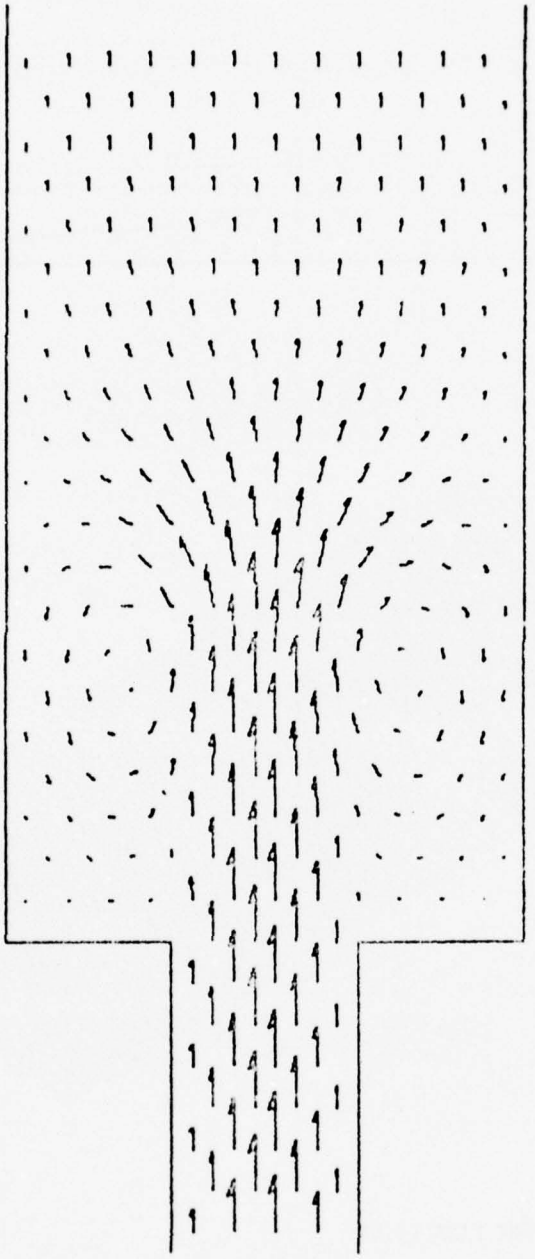
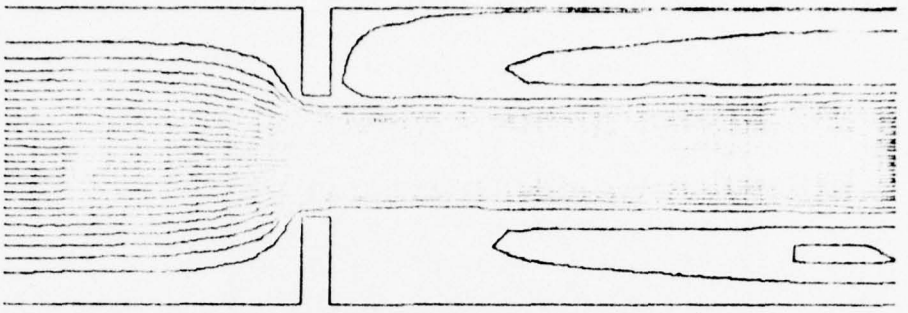
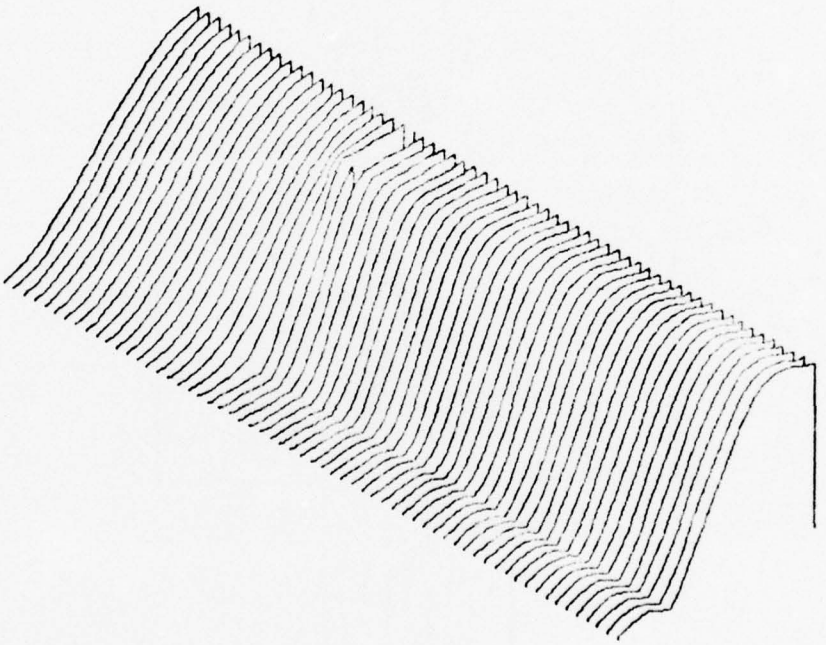


Fig. 118: Velocity Vector Plot for Idealized Flow into Simplified Atrium Section of the Heart (One Time Frame)



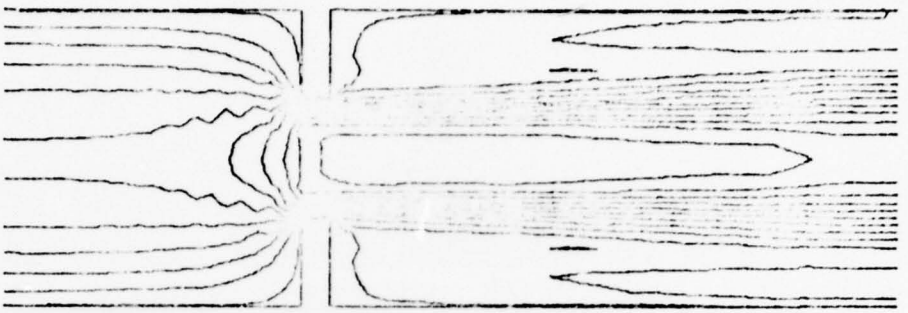


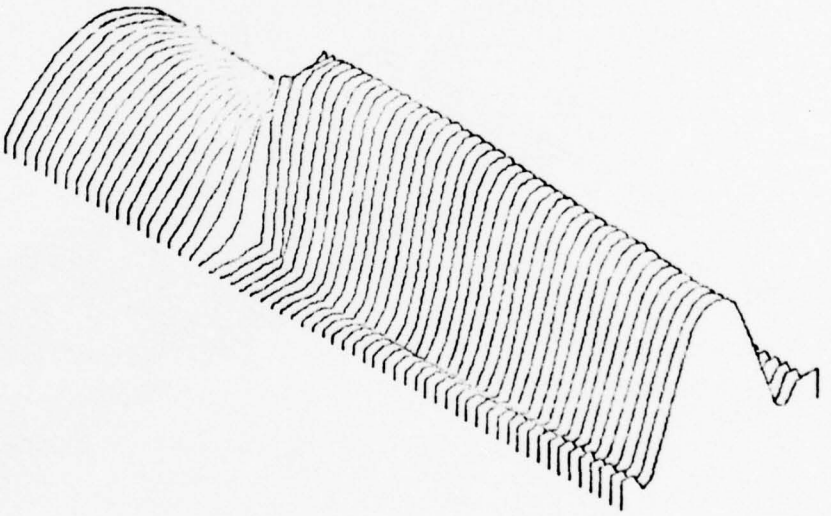
STREAM FUNCTION

R= 400.

FRAME NO. 10.

Fig. 119: Stream Function Plot for Two Arbitrary Shaped Flow Areas (Amenable to Particular Applications such as Two Cusped Leaflet)



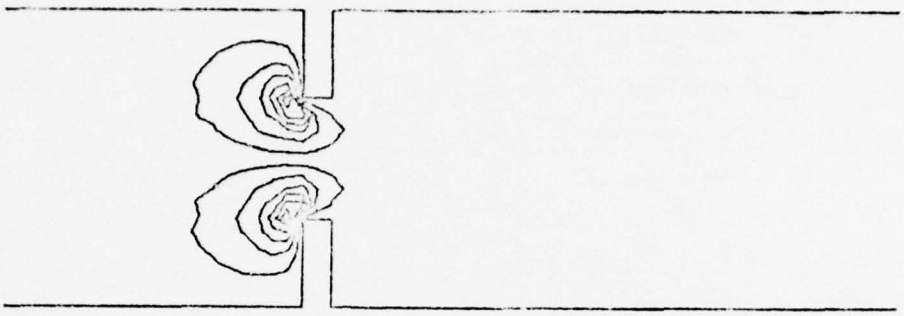


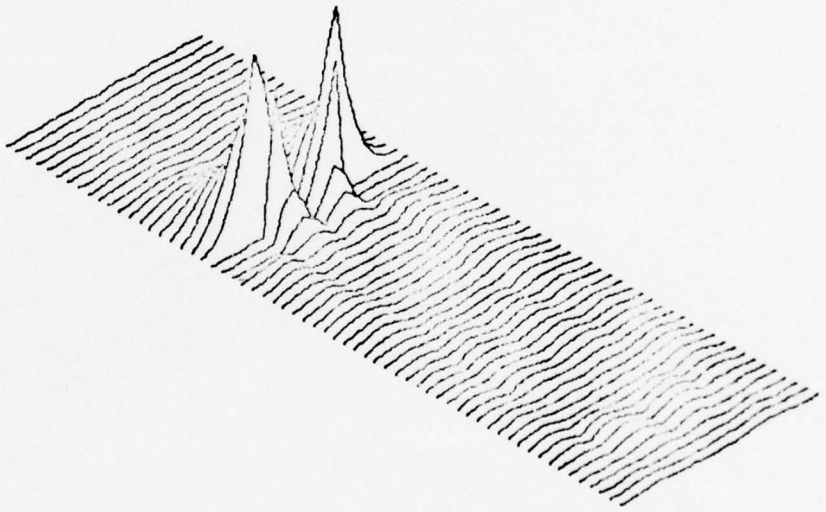
U VELOCITY

R= 400.

FRAME NO. 10.

Fig. 120: Horizontal Velocity Function Plot for Two Arbitrary Shaped Flow Areas (One Time Frame)



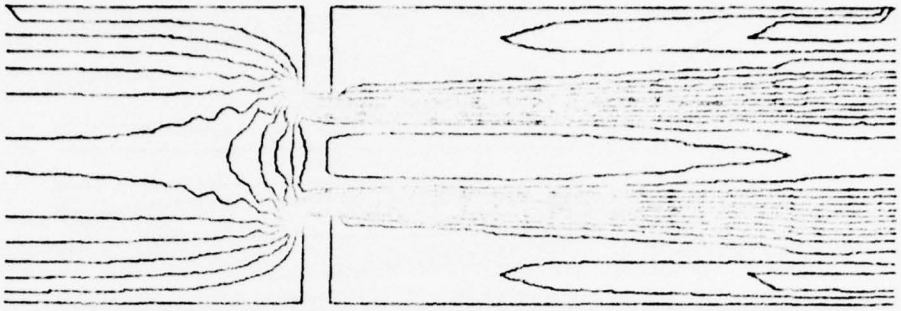


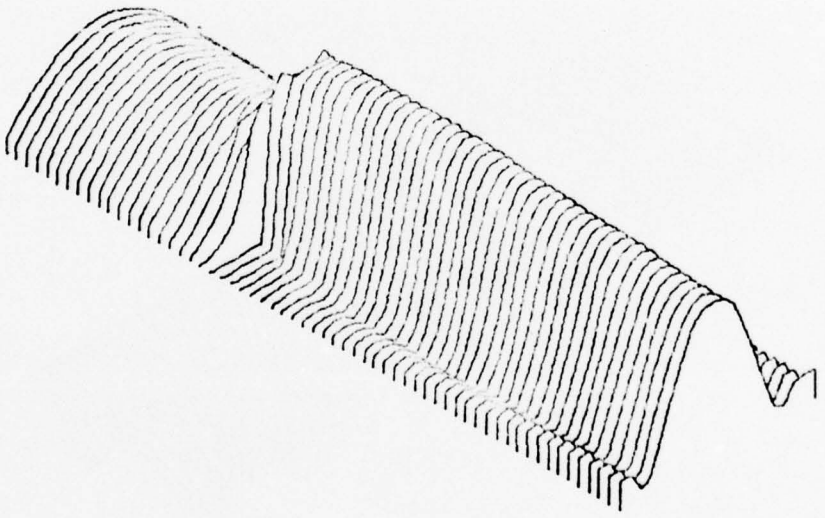
V VELOCITY

R= 400.

FRAME NO. 10.

Fig. 121: Vertical Velocity Function Plot for Two Arbitrary Shaped Flow Areas (One Time Frame)



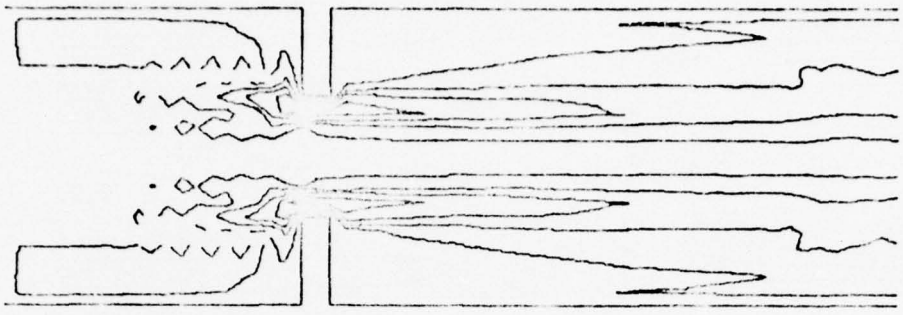


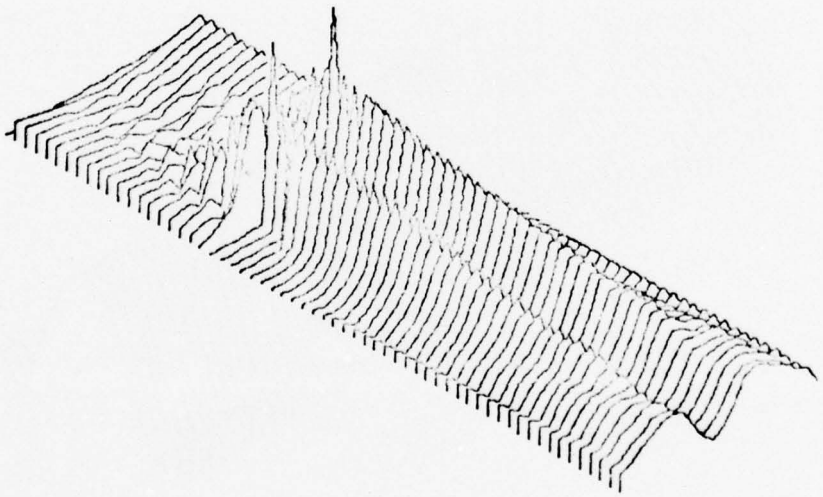
TOTAL VELOCITY

R= 400.

FRAME NO. 10.

Fig. 122: Total Velocity Function Plot for Two Arbitrary Shaped Flow Areas (One Time Frame)





VORTICITY FUNCTION

R= 400.

FRAME NO. 10.

Fig. 123: Vorticity Function Plot for Two Arbitrary Shaped Flow Areas (One Time Frame)

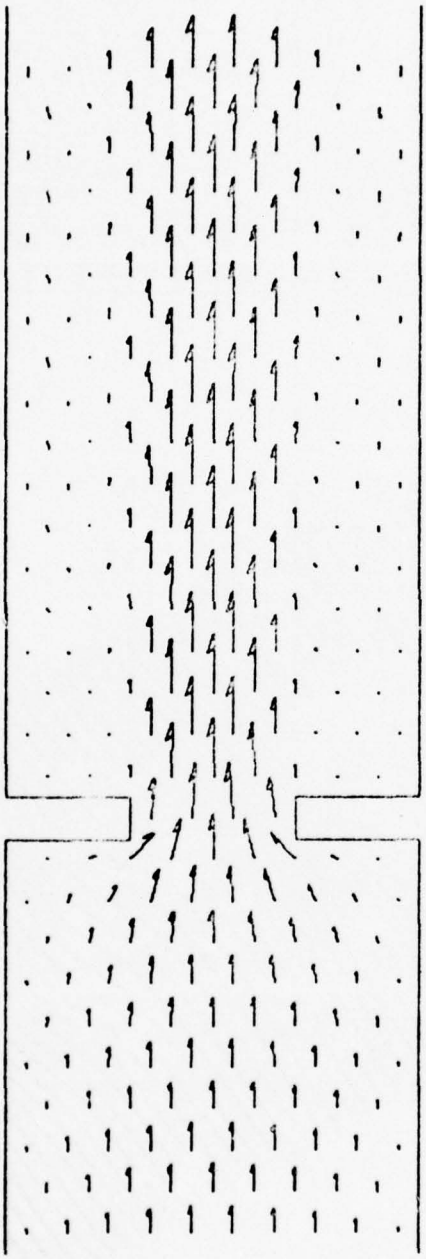


Fig. 124: Velocity Vector Plot for Two Arbitrary Shaped Flow Areas (One Time Frame)

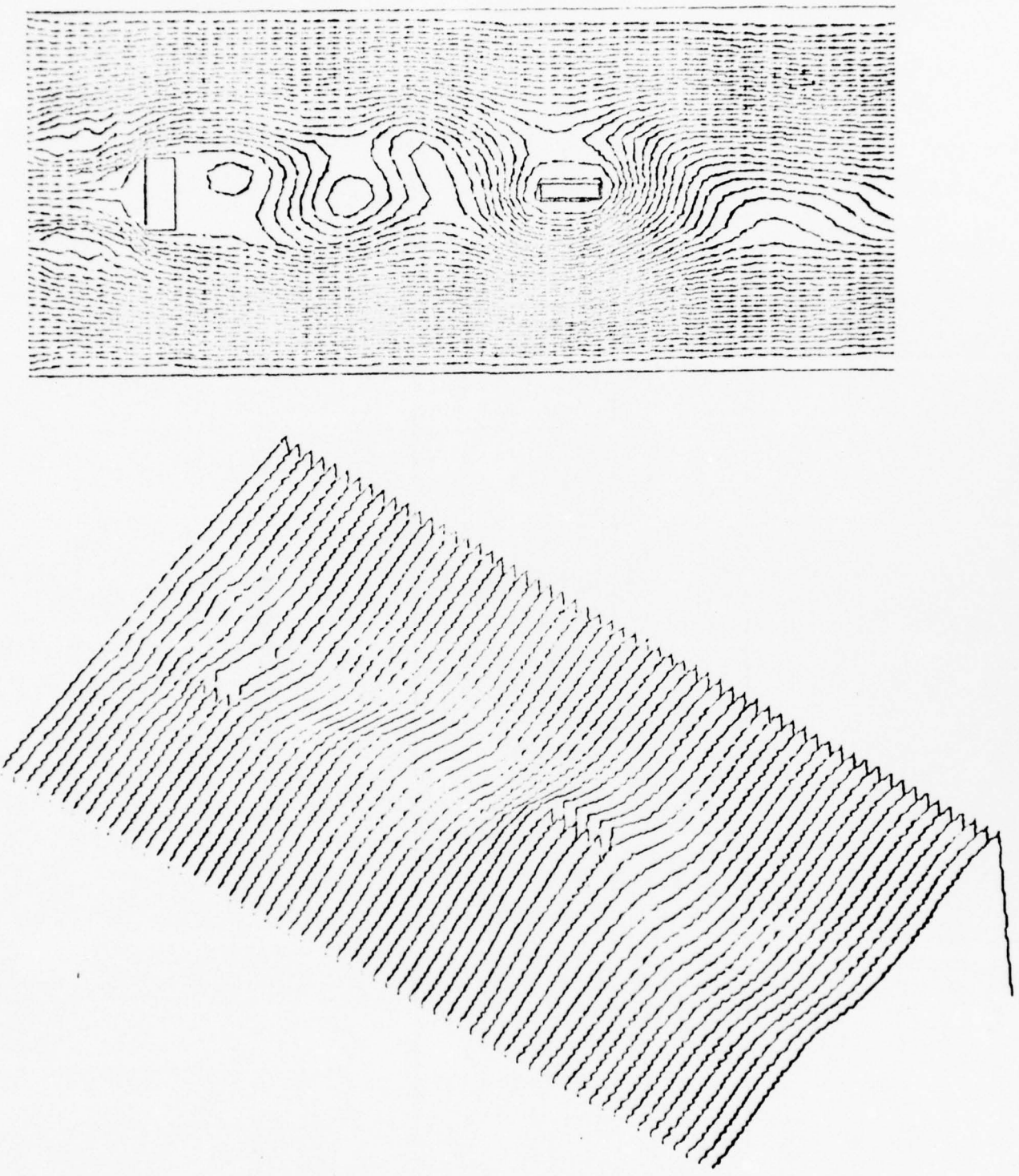


Fig. 125: Stream Function Plot for Two Obstacles in a Flow Field
(One Time Frame)

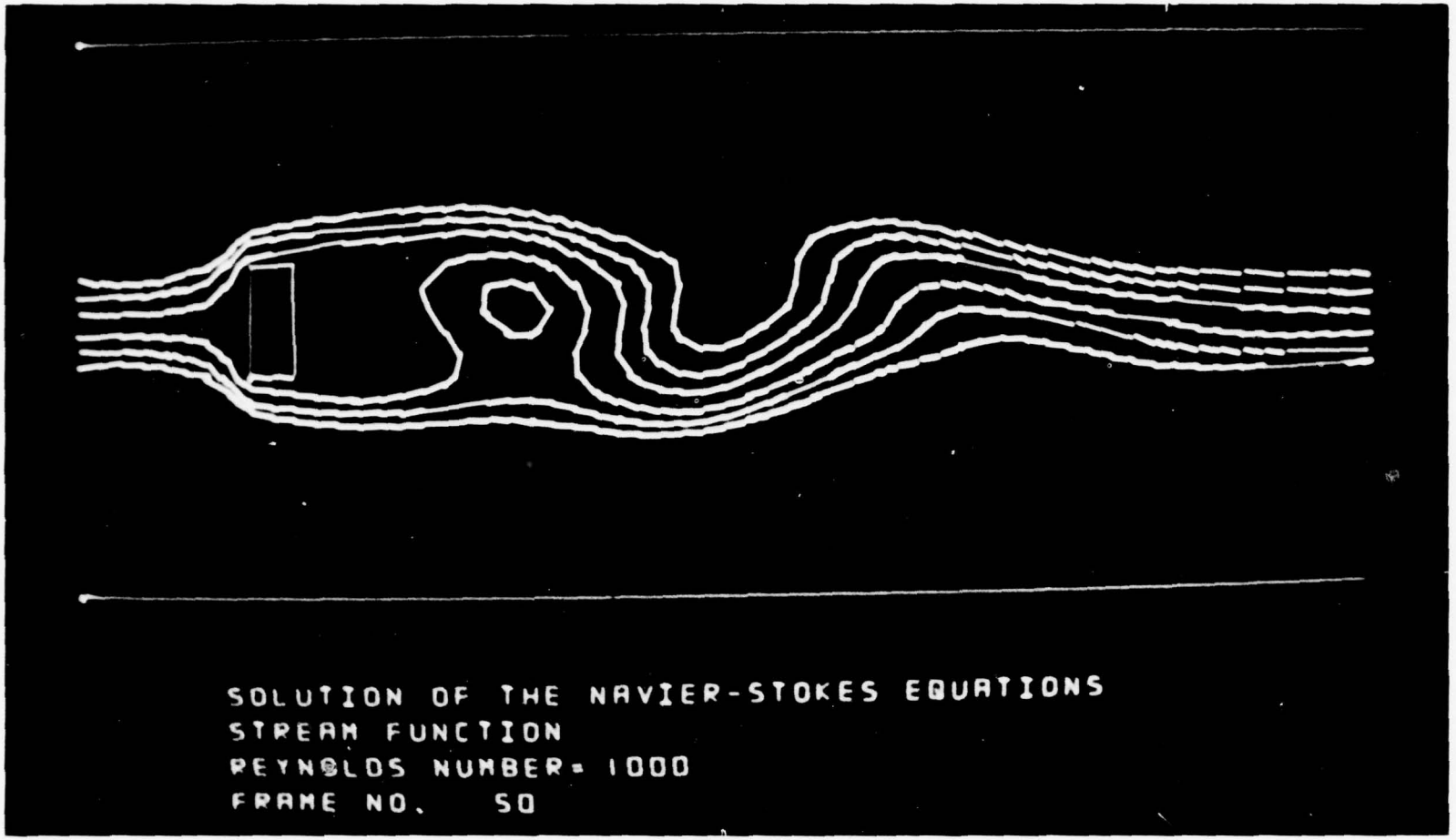


Fig. 126: Photograph from IDI Console Screen Showing Stream Function Plotted for Detailed Study of Internal Core of Flow.

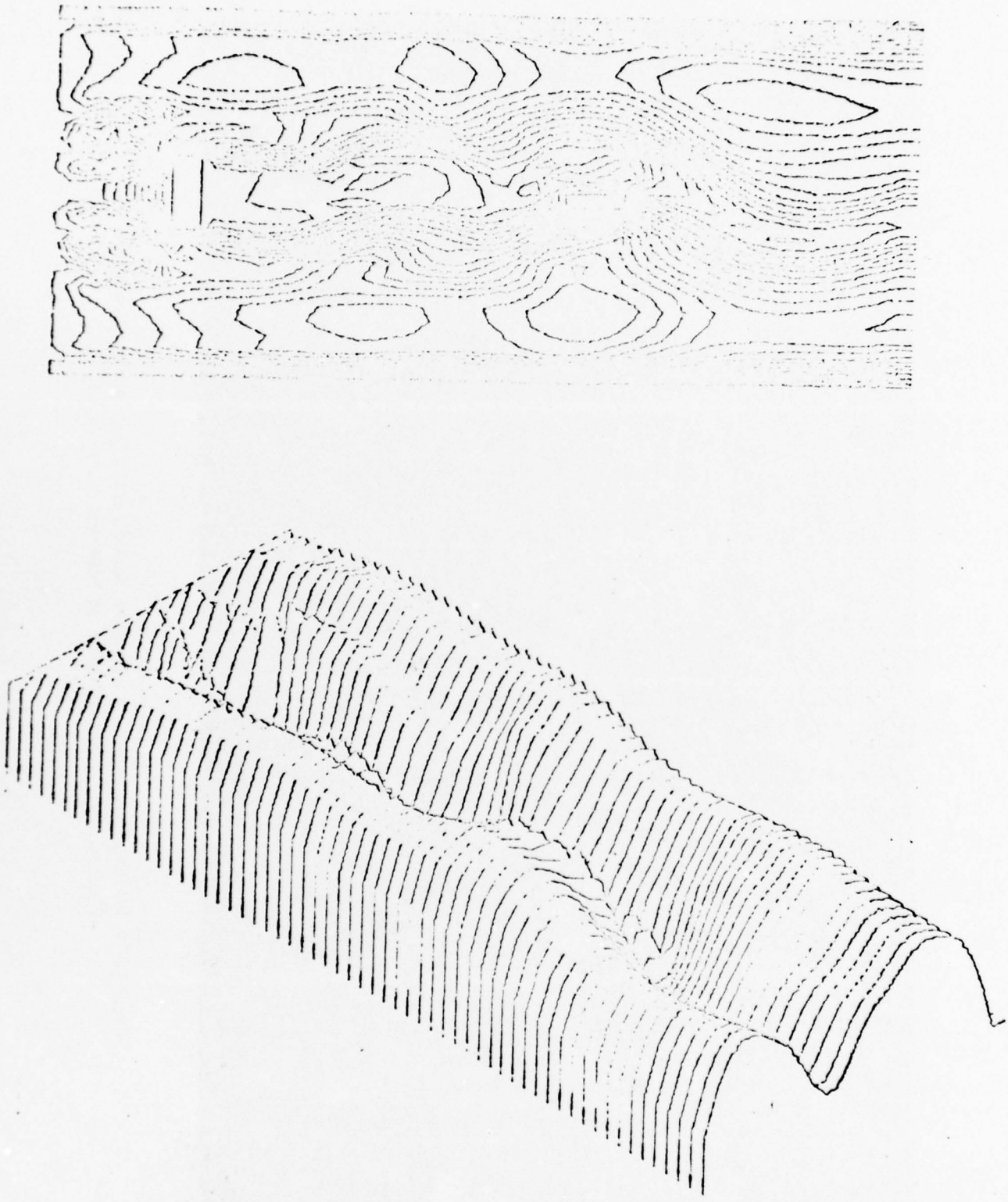
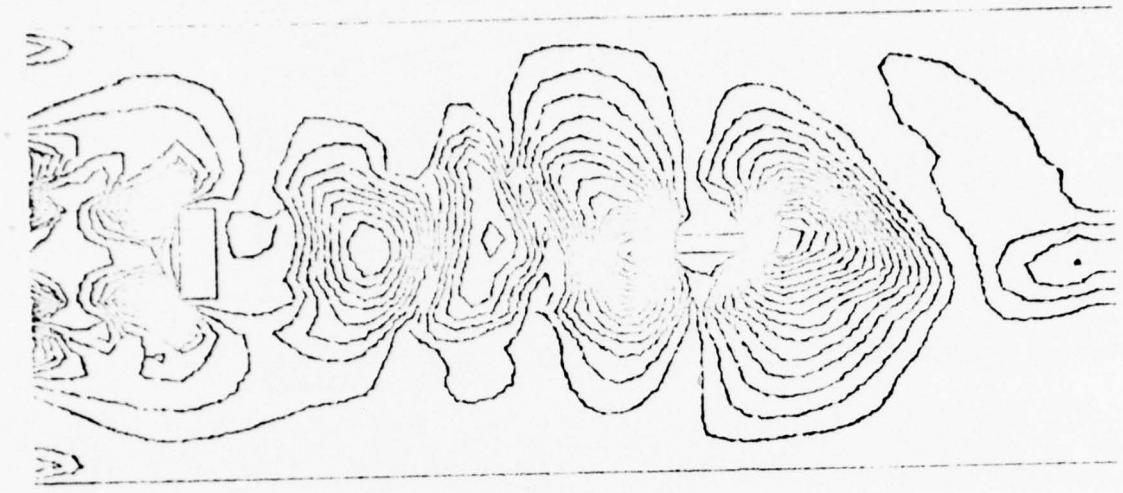


Fig. 127: Horizontal Velocity Function Plot for Two Obstacles in a Flow Field (One Time Frame)



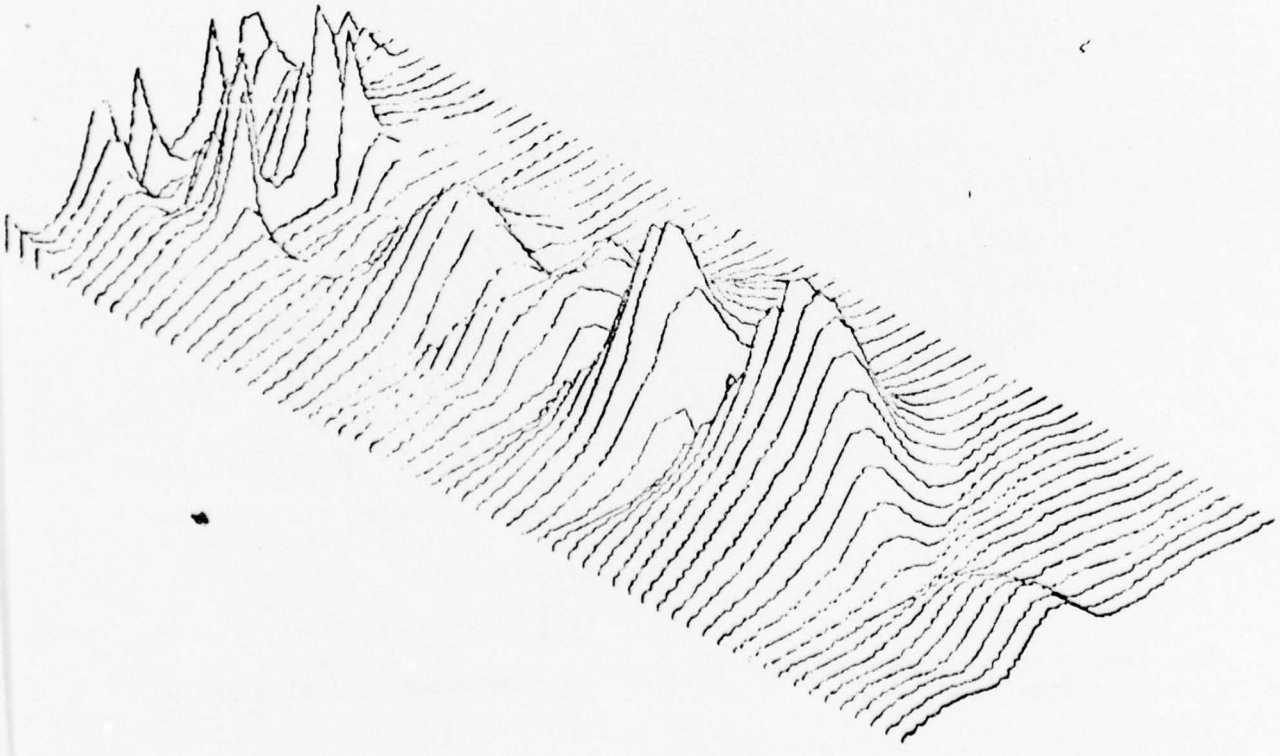
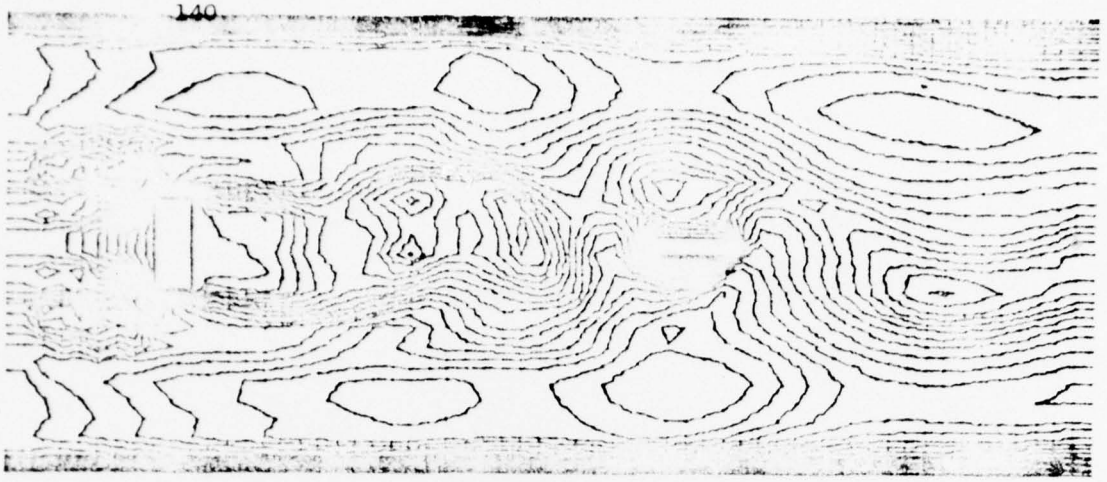


Fig. 128: Vertical Velocity Function Plot for Two Obstacles in a Flow Field (One Time Frame)

V VELOCITY AT TIME T= 0.00625

R= 2000.



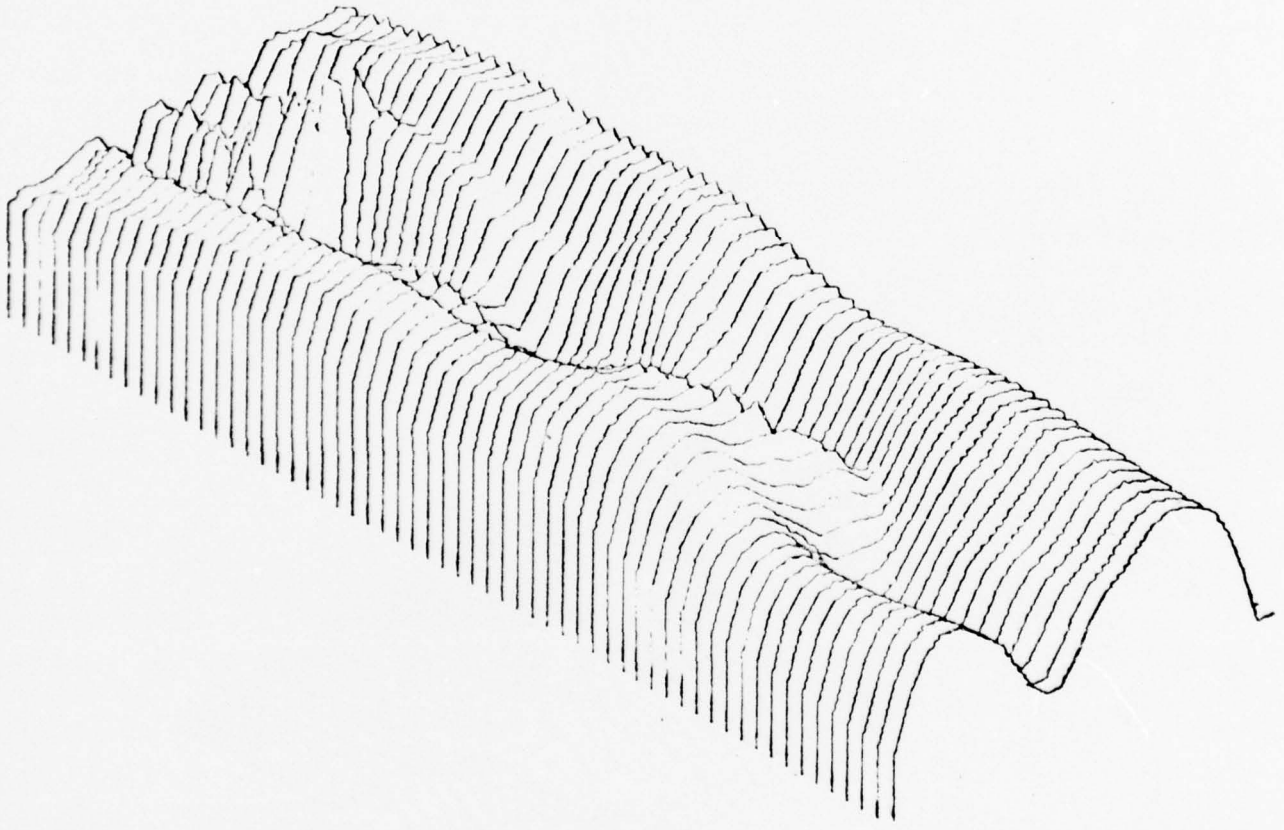


Fig. 129: Total Velcoity Function Plot for Two Obstacles in a Flow Field (One Time Frame)

TOTAL VELOCITY AT TIME T= 0.00625

R= 2000.

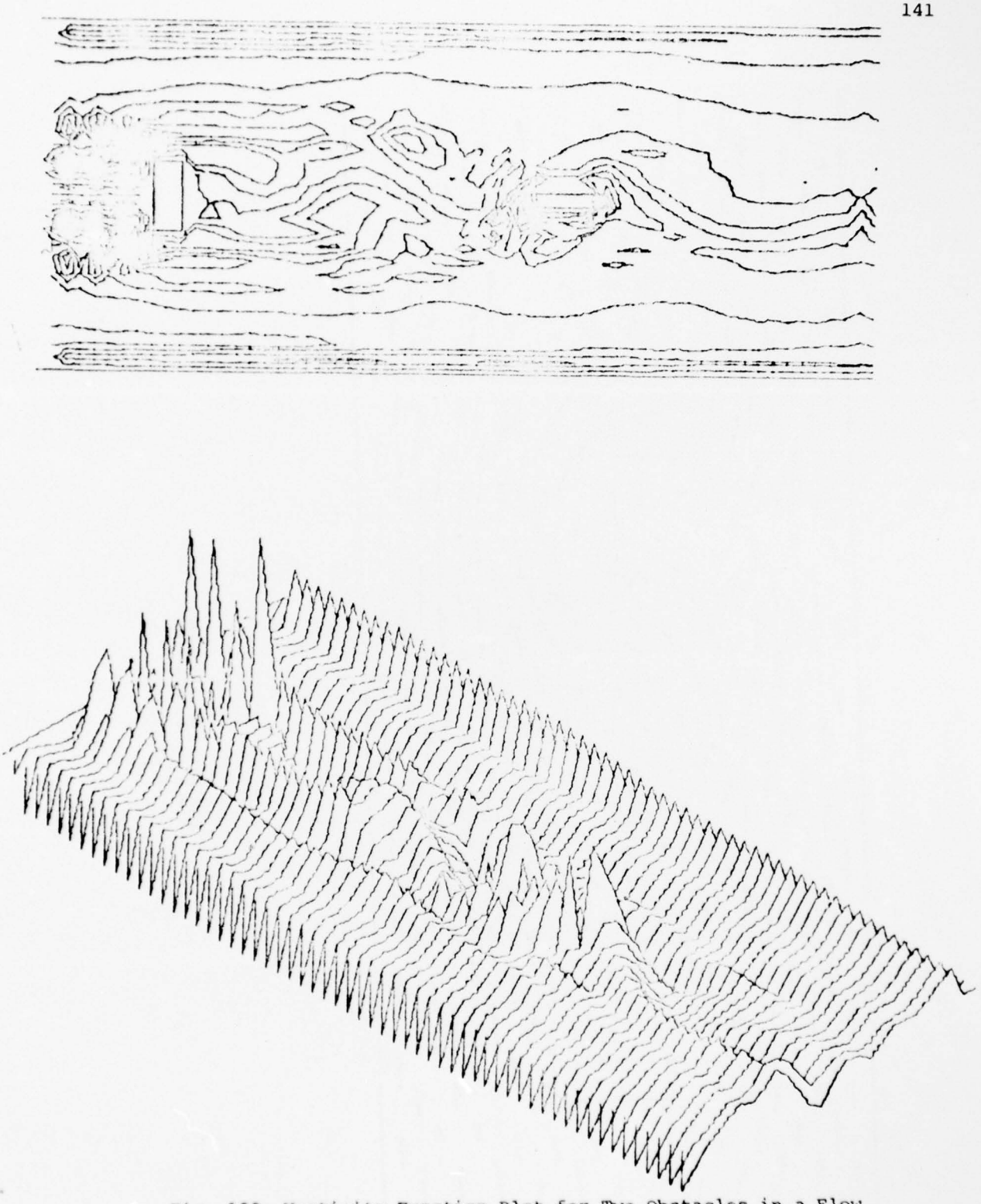


Fig. 130: Vorticity Function Plot for Two Obstacles in a Flow Field (One Time Frame)

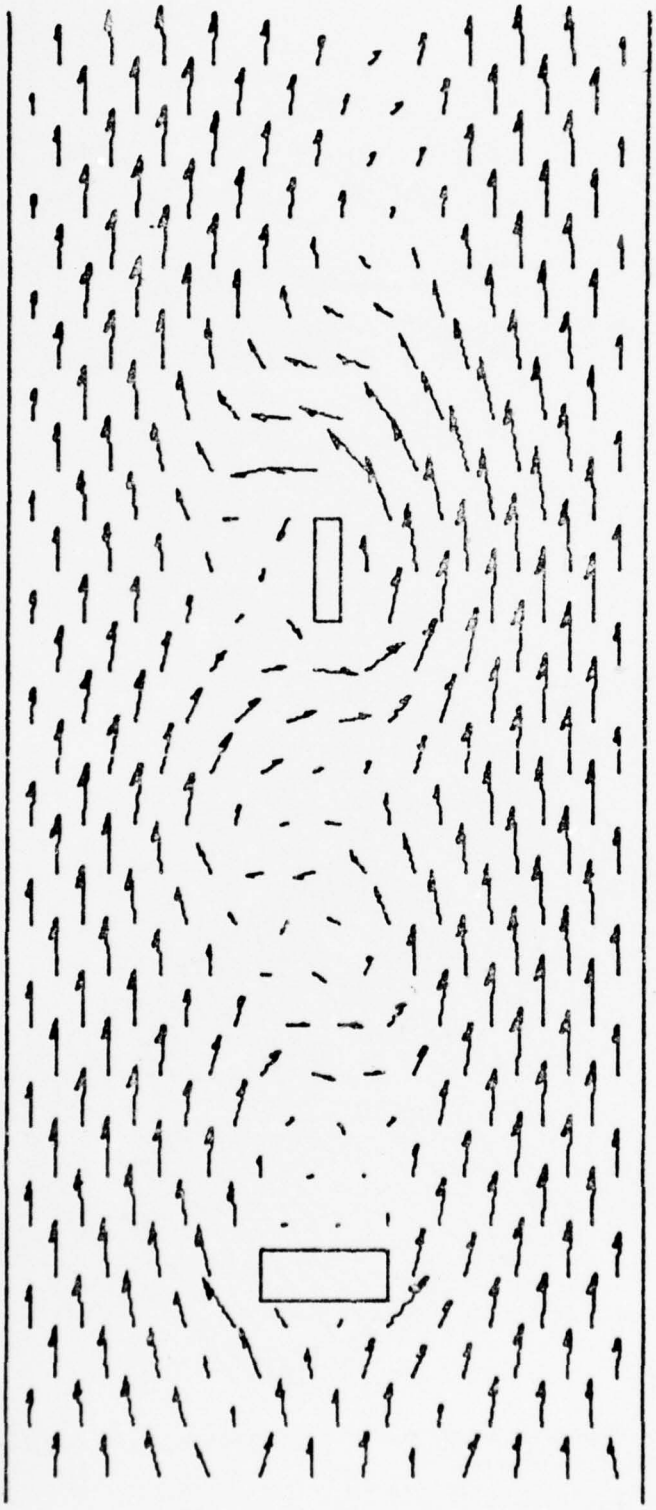


Fig. 131: Velocity Vector Plot for Two Obstacles in a Flow Field
(One Time Frame)

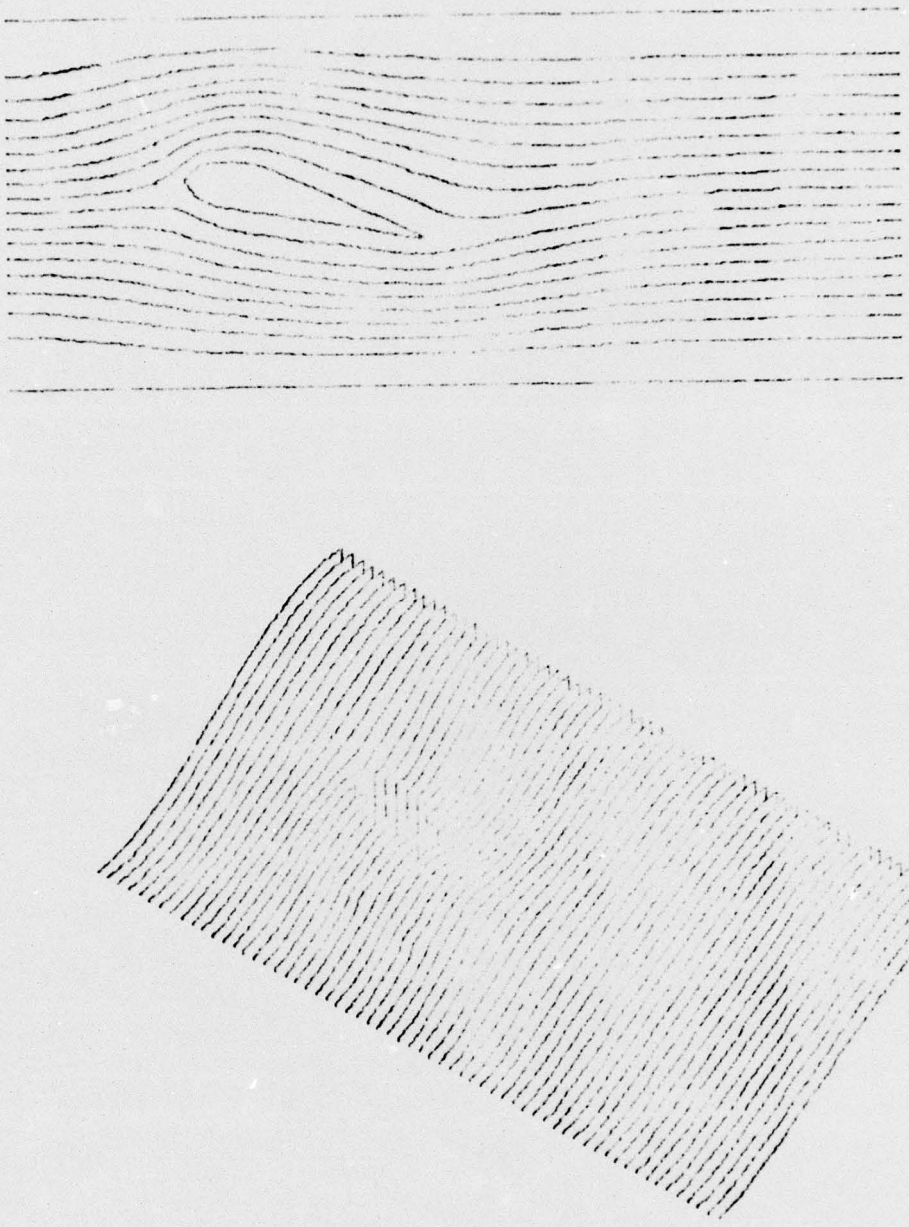


Fig. 132: Stream Function Plot for An Arbitrary Aircraft Wing Shape (One Time Frame)

PRECEDING PAGE BLANK-NOT FILMED

CHAPTER VI

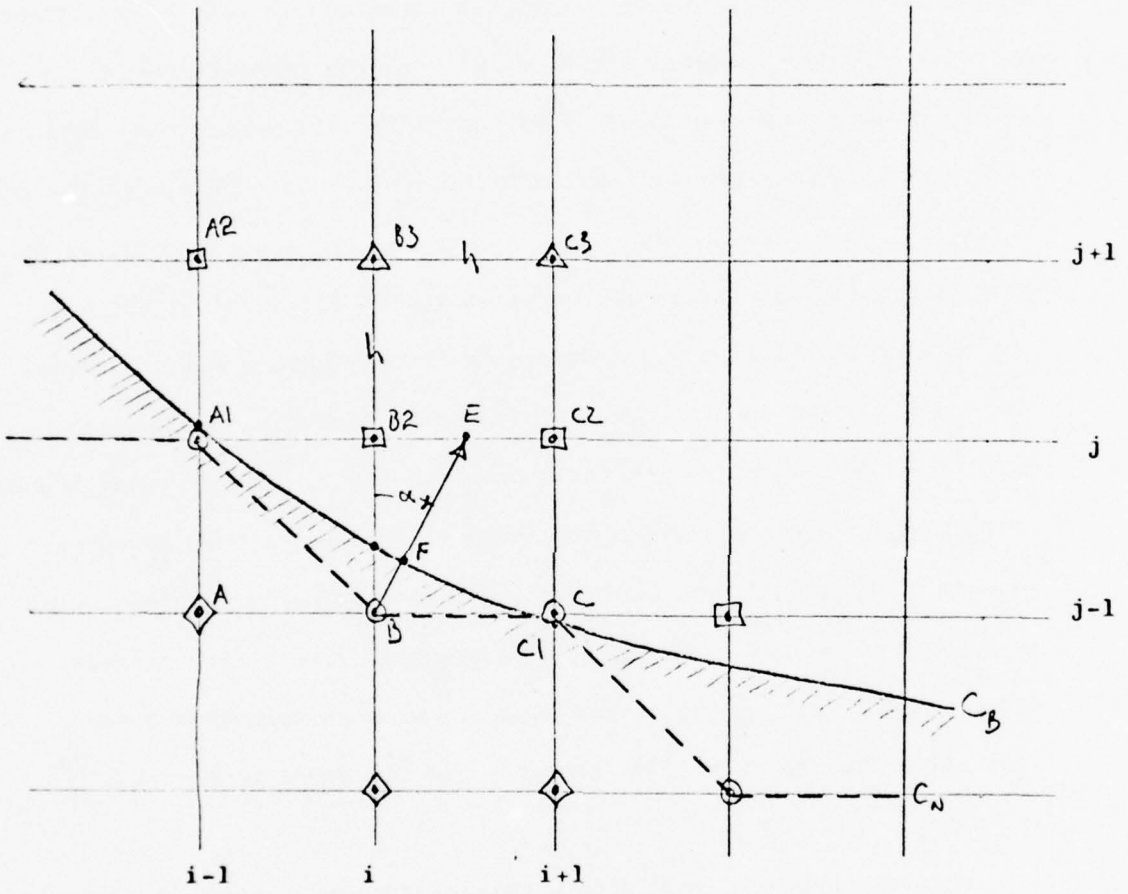
PRESENTATION OF A CURVED BOUNDARY ALGORITHM

The main objection to the previous attempts at a numerical solution to the general set of the Navier-Stokes equations is that they are not general. Previous investigators, as well as the present ones up to this point, assumed that the boundary of the finite difference mesh constituted either a fluid boundary or a solid boundary that coincided with the computational mesh lines. Many problems are ignored and the computer program is greatly simplified if one tries to fit all physical boundary configurations to the lines of a square finite difference mesh, however, the usefulness of such a computer program is doubtful. In general, most flow, design, and/or obstacle problems require an irregular boundary configuration. The algorithm presented in this section is an attempt to form the required numerical routine for allowing curved boundaries into the desired two dimensional flow problem. The finite difference equations for forming the curved boundary problem were thus formed. Specialized finite difference concepts are discussed by Forsythe [54], Salvadori [55], and Todd [56].

The computational mesh with stencils of unequal arms, as discussed in section B2, was employed, as well as the finite difference forms of section B3. As before, the following definitions were set;

a. Let the superscript k denote the value of the variables at the kth iteration.

b. Let the subscripts A,B,C denote the boundary points, adjacent interior points, and remaining interior points, respectively, of the node points on the finite difference mesh grid, as seen in figure 133. For example ψ_{B2}^{k+1}



Adjacent interior point (type 0)

Interior point (type 0)

Node boundary point (type 1)

Fxterior node point (type 2)

Fig. 133: Computational Mesh with Curved Boundary Involved.

denotes the value of the stream function evaluated at the k + 1th iteration on the point (i,j) where (i,j) corresponds to a boundary point.

The algorithm follows the previous one up to a certain point. For the sake of continuity, the totality of steps are shown.

1. The values $\psi_{B,C}$, $\omega_{B,C}$ were estimated by extrapolation from past values.
2. The improved values of ω_C were calculated from

$$\omega_t = \nabla^2\omega - \psi_y\omega_x + \psi_x\omega_y$$

(in finite difference form, with the right hand side averaged between level k and k + 1).

Again, the method of calculation was the Peaceman-Rachford alternating direction method. Smoothing was necessary for higher Reynolds numbers.

3. The stream function values on the adjacent interior points were improved by the following formulae;

$$\psi_{B2} = \psi(x_B, y_B, t_{n+1}) + h\psi_y(x_B, y_B, t_{n+1}) + \frac{h^2}{2}\psi_{yy}(x_B, y_B, t_{n+1}) + O(h^3) \qquad (59)$$

Since $\psi_{yy} = -2\omega - \psi_{xx}$ then

$$\psi_{yy}\Big|_B = -2\omega_B - \frac{1}{h^2}(\psi_A - 2\psi_B + \psi_C) \qquad (60)$$

But ω_B is a boundary point and was not known, therefore extrapolation from known interior points was required, hence $\omega_B = 2\omega_{B2} - \omega_{B3}$

and the following improvement formula was required;

$$\psi_{B2}^{k+1} = \psi_B^k + h\psi_{yB}^k + \frac{h^2}{2}\left[- 4\omega_{B2}^k + 2\omega_{B3}^k - \frac{1}{h^2}\left(\psi_A^k - 2\psi_B^k + \psi_C^k\right)\right] \tag{61}$$

Since the boundary conditions $\psi_x = \psi_y = 0$ were necessarily satisfied everywhere on a fixed solid boundary, then if the boundary is parallel to the mesh,

$$\psi_{yB}^k = 0$$

and equation (61) reduced to

$$\psi_{B2}^{k+1} = \psi_B^k + \frac{h^2}{2}\left[- 4\omega_{B2}^k + 2\omega_{B3}^h - \frac{1}{h^2}\left(\psi_A^k - 2\psi_B^k + \psi_C^k\right)\right] \tag{62}$$

If the boundary is curved, however, as depicted in figure 133 (line C_B) then one must make some additional calculations. Consider the mesh node point (i,j) in figure 133. The net boundary is shown by the dashed line C_N. At each node of C_N, a perpendicular was drawn through C_B and extended until it crossed a link or node of the net in the interior. For example, the perpendicular from B meets C_B at F, and meets the link from B_2 to C_2, involving the angle α.

A value for ψ_E was determined by linear interpolation between ψ_{B2} and ψ_{C2}. A "normal" derivative for the linear interpolation was then obtained for the net function ψ, by

$$\frac{\psi_E - \psi_B}{h/\cos\alpha} \tag{63}$$

Equating (63) to the normal derivative at F, one obtained the desired equation for ψ_B. For example, expressing ψ_E in terms of ψ_{B2} and ψ_{C2} by

$$B_2 \cdot E = h \tan \alpha, \quad E \cdot C2 = h(1-\tan \alpha) \qquad (64\ a,b)$$

then

$$\psi_{B2}(\cos \alpha - \sin \alpha) + \psi_{C2} \sin \alpha - \psi_B \cos \alpha =$$

$$h\psi_n \Big|_F = 0 \qquad (65)$$

hence,

$$\psi_B = \frac{1}{\cos \alpha} \left[\psi_{B2} (\cos \alpha - \sin \alpha) + \psi_{C2} \sin \alpha \right] \qquad (66)$$

The equation for the adjacent interior node B2 that was analogous to equation (61) was then

$$\psi_{B2}^{k+1} = \psi_B^k + \frac{h^2}{2} \quad - 4\omega_{B2}^k - 2\omega_{B3}^k - \frac{1}{h^2}$$

$$(\psi_A^k - 2\psi_B^k + \psi_C^k)$$

where ψ_A^k, ψ_B^k, and ψ_C^k are pseudo-boundary points that were obtained by extrapolation from the known values, and where ψ_A^k, ψ_B^k, and ψ_C^k all satisfied equation (66).

4. The improved values of ω_B were calculated by

$$\text{NEW } \omega_B = SF \cdot (\text{OLD } \omega_B) + (1-SF)\ T$$

where $T = -1/2\nabla^2\psi \Big|_B$ and where SF is a smoothing factor.

5. Test for convergence; if not convergent one returned to step 1.

A. Discussion for Atherosclerosis Study

Examples of computer graphic flow plots follow. All are equivalent to the original "experiments" as performed on the display scope, however, the alternate program sub-routine for employing the Gerber plotter was used to allow the particular one time-frame results shown. Parametric flow solutions are first shown for particular vascular situations in the renal section and in the abdominal aortic bifurcation section. This group is then followed by graphic results formed for certain artificial heart valve situations.

In most evaluations the plots for the stream function, horizontal velocity component, vertical velocity component, total velocity function, vorticity function and velocity vector are shown. All have their surface and isometric values formed. Although details of concern to each plot have not been pursued, they are available for future study. One can, however, by qualitative viewing, infer some interesting points. For example, figures 135 and 137 show the typical laminar flow front, to be expected at the Reynolds number employed in those results. In turn, the isometric plots of figure 135 and 136 combine to form the isometric plot in figure 137. Flow is from left to right. In some instances the horizontal velocity plot appears to be equivalent to the total velocity plot in isometric view. Qualitatively this is equal to saying that the vertical velocity component was extremely small in value. Figure 138 shows heavy concentrations of vortices (forerunners of turbulence) rapidly appearing at the entrances to the smaller arterial channels, with interacting vortices after a right angle turn from the main channel flow. The equivalent isometric picture shows the sudden rises in vortex magnitudes at these clustered positions. The isometric plot allows quantitative

evaluation since the vortex peaks arise from a flat background that constitutes a baseline value.

Note that figure 134 through 139 did not require the curved boundary algorithm. Figure 140, as an example of the first use of the curved boundary algorithm, does show a heavy vortex concentration within the upper side artery leading to one kidney from the mesentery artery theoretical equivalent. Depending upon the particular time frame employed, if viewed in the light of a Von Karman vortex street likeness, that particular vortex could have appeared in the other side artery.

Figures 141 through 145 depict computer graphic results as applied to the more natural situation of non-orthogonal side channels emanating from the main artery in the renal section. The important point is that the side arteries are programmed to angle at any desired inclination, just as clinical results show for various individuals. The upper side artery was arbitrarily placed facing upstream and the lower one was faced downstream to the general flow.

Figure 141 bears out physical concepts in that the streamlines are shown closer in proximity upstream of the side arteries, therefore, showing lessening of flow energy downstream as flow is drained off in the side channels. Figure 142 shows the typical parabolic type front for laminar flow, changing to a typical turbulent front in the main flow region between the side arteries. Figure 144 shows a proper physical amalgamation of figures 142 and 143, the usual horizontal and vertical velocity components together forming a total velocity value in the isometric view. Figure 145 shows the vorticity function which then became the subject of detailed sequential study within the particular portion of the backward facing side artery region, resulting in figure 146. Figure 145's isometric plot particularly shows the extremely

high vortex peaks in the upper channel, which was to be expected as the blood flow radically changed direction. Realizing this, the Reynolds number was continually changed in the montage components of figure 146 to allow a transition from a slow initial rate of flow to slightly above an average blood flow velocity value in that particular area. One then notes the formation of a vortex in such a velocity regime and if the observer views the small triangular shaped vortex on the anterior wall, the breakup and reformation of stream function lines in amplifying this vortex are clearly noticed.

Figure 147 depicts the various plots as formed for an idealized atherosclerotic lesion formed on a vascular wall. Present work, involving stress analysis at the vascular wall, but not reported here, will certainly incorporate the program subroutines that formed this particular picture. This study area is to be extended and one will attempt to understand the growth and interplay of flow parameters as the plaque grows.

Figures 148 through 153 show the stream function, horizontal velocity component, vertical velocity component, total velocity, vorticity function and velocity vector for a particular Reynolds number value at the abdominal aortic bifurcation region. Figure 152 may be singled out as particularly interesting and for comparison to figure 154.

Note that figure 154 is a diagrammatic representation of sites of predilection for the occurrence of atherosclerosis. Wesolowski [23] and other vascular surgeons, using hydraulic models of various configurations, and from observations at operations and by arteriography, have noted that turbulence occurs at each site of such predilection. Comparison then, of figure 152 with figure 154 shows that the theoretical vorticity function plot (vorticity, the forerunner of turbulence), is equivalent for site positioning with observed sites of turbulence and eventually atherosclerosis site prediction.

Figures 155 through 158, and 160 and 161, view the computer graphic results, at a particular instant of time and for an arbitrary Reynolds number, for blood flow within the renal and abdominal aortic bifurcation sections together. The physical flow values, consistent with those of an average, healthy, young male, will be introduced into such a computer model in the near future. If one compares figure 160 with figure 152, it appears that the vortices at the bifurcation are shifted downstream in figure 160 and added lengths of vortex motion are noted. Such results are hypothesized to be due to the role played by the renal section as an upstream addition. This concept is discussed by Greenfield [57, 58]. Figure 160, when compared to figure 161, shows that orthogonal side arteries at the renal section do have somewhat different vortex motion than askew arteries. In addition, however, the angled renal arteries appear to produce heavier vortex motion within the bifurcated channels downstream.

B. Curved Boundary Computer Graphic Displays. (Vascular)

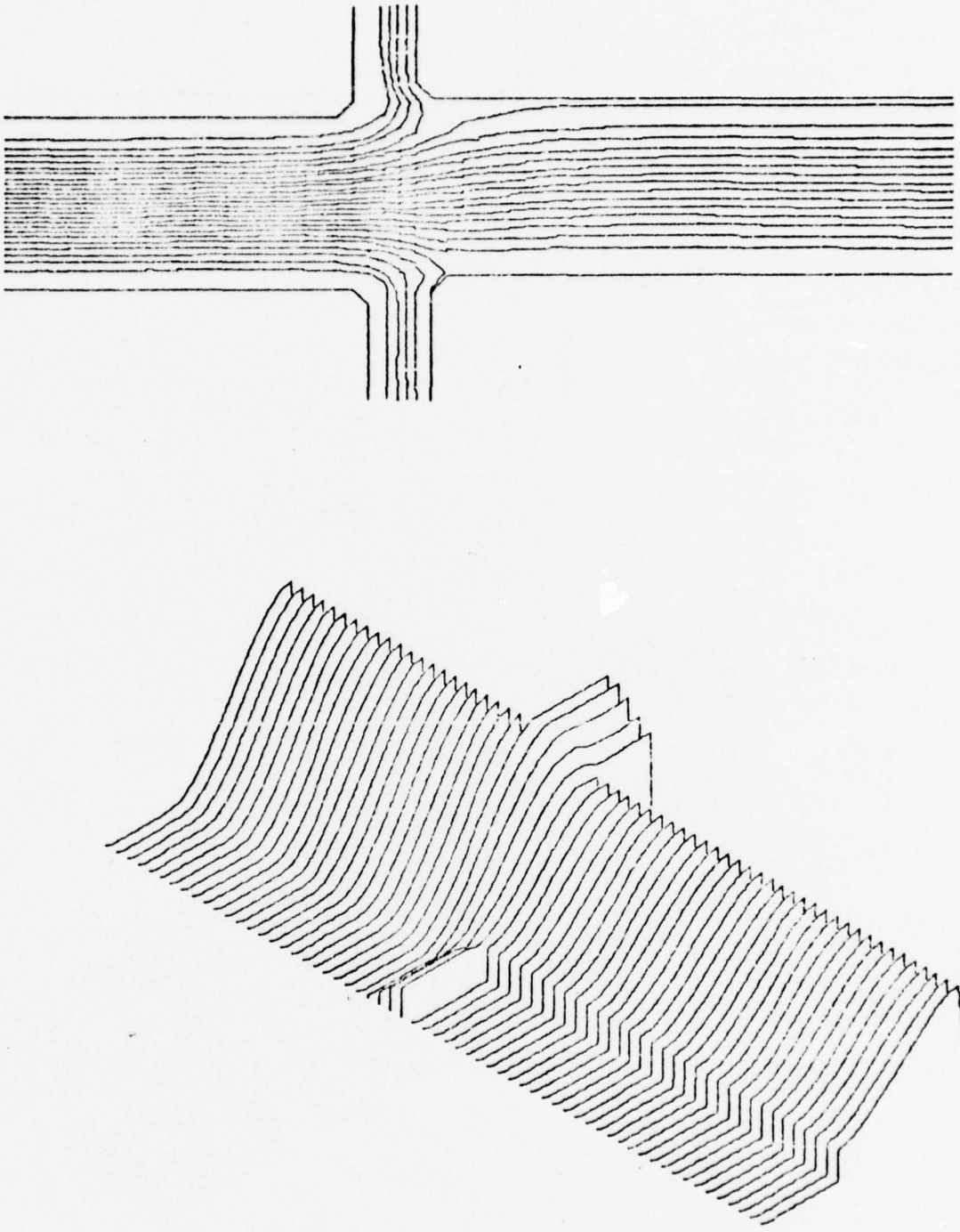
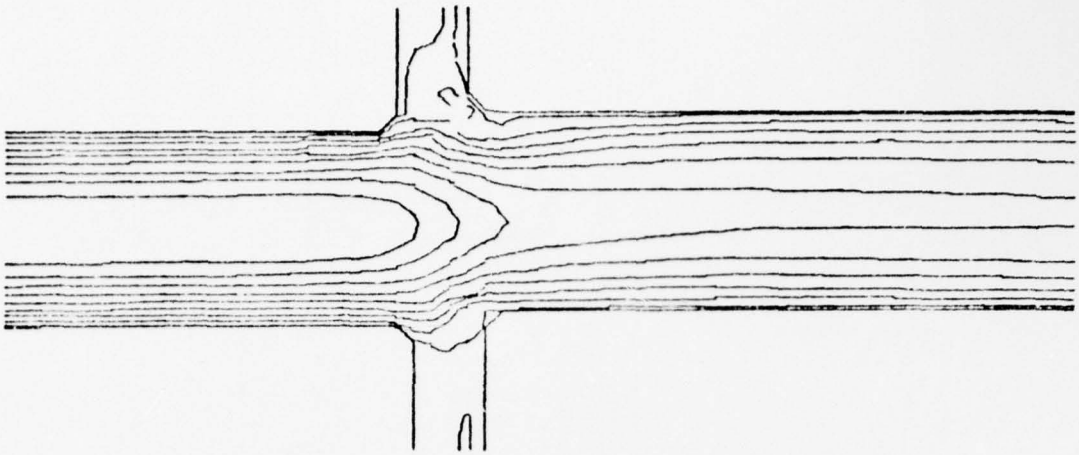
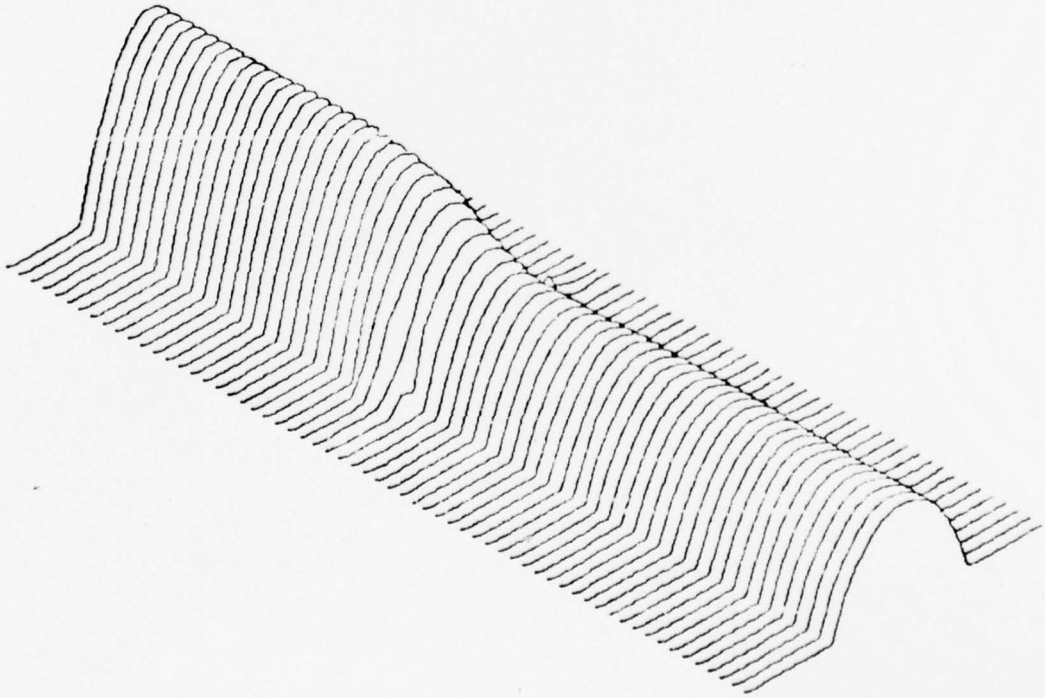


Fig. 134: The Stream Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries; One Time Frame)





U VELOCITY

R= 500.

FRAME NO. 10.

Fig. 135: Horizontal Velocity Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries; One Time Frame)

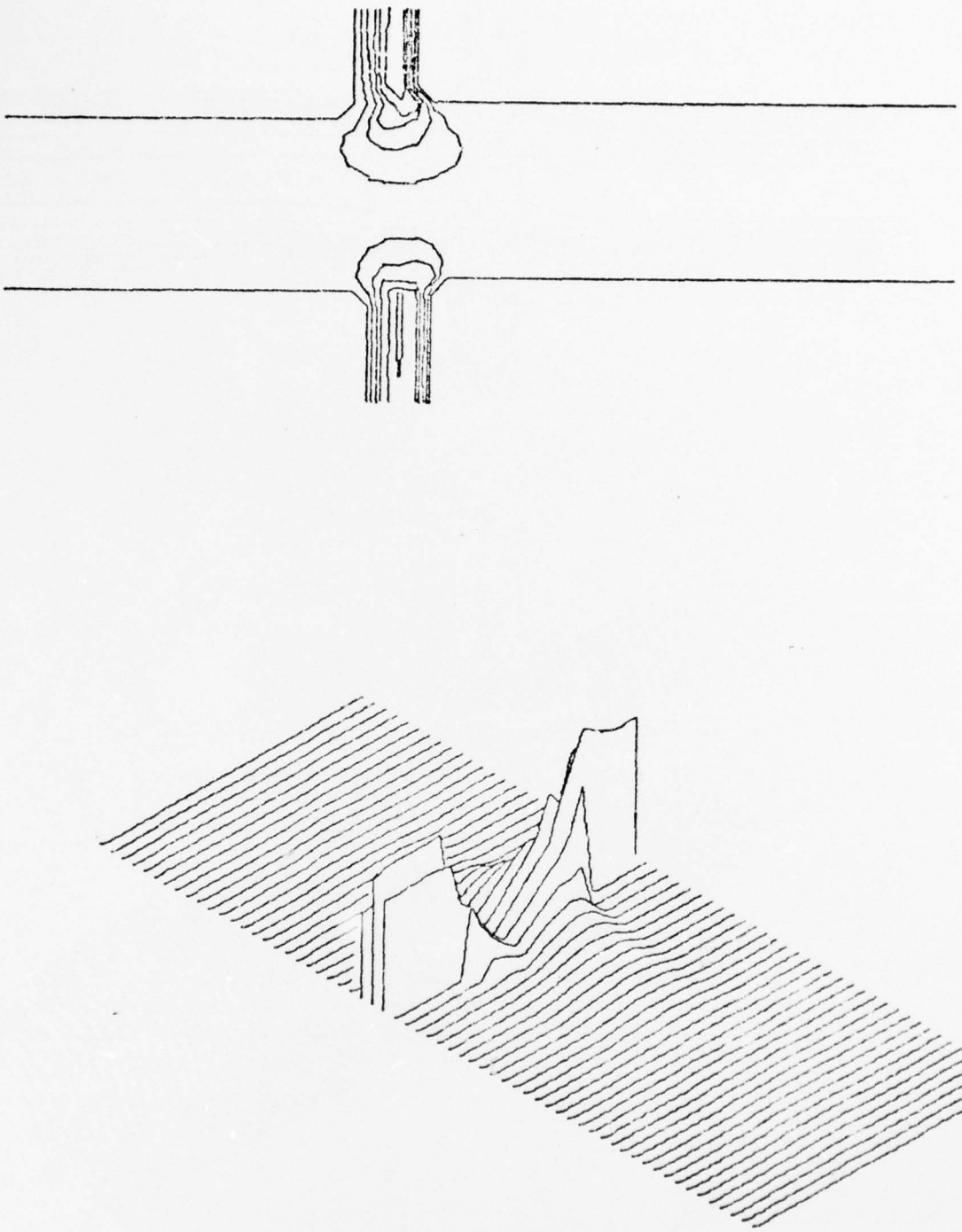
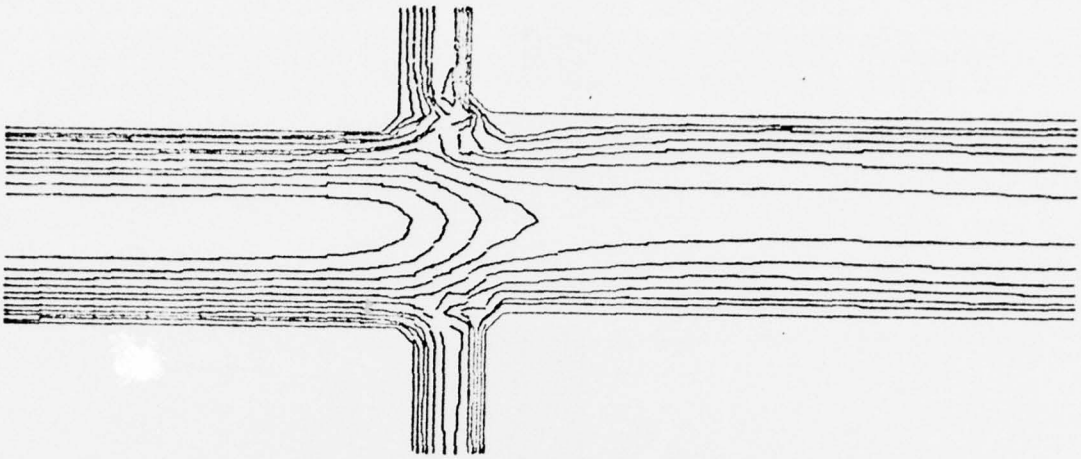
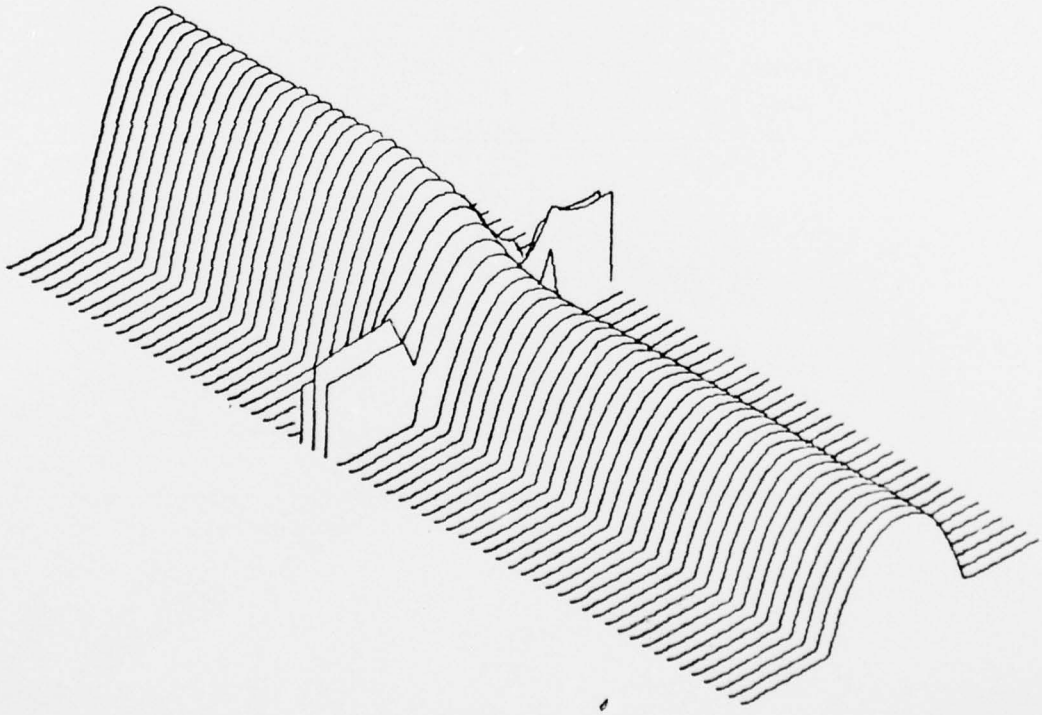


Fig. 136: Vertical Velocity Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries; One Time Frame)



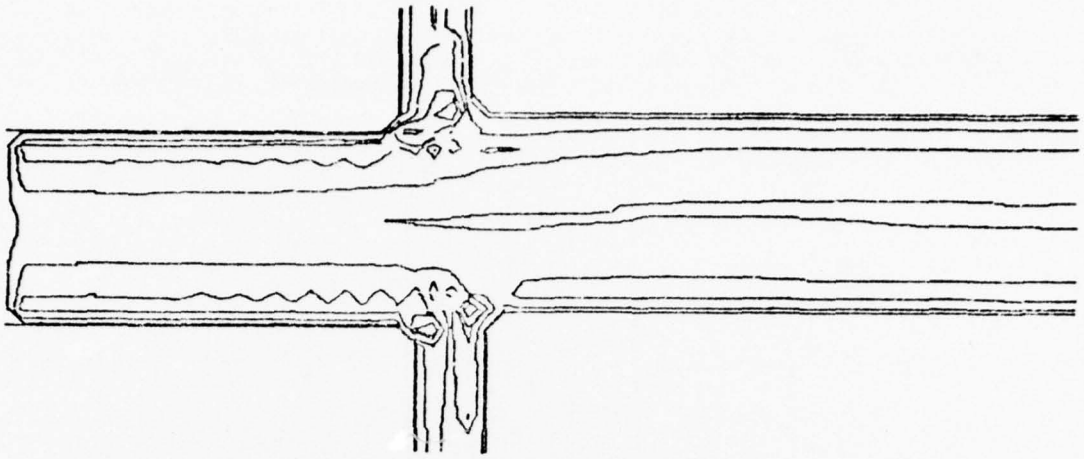


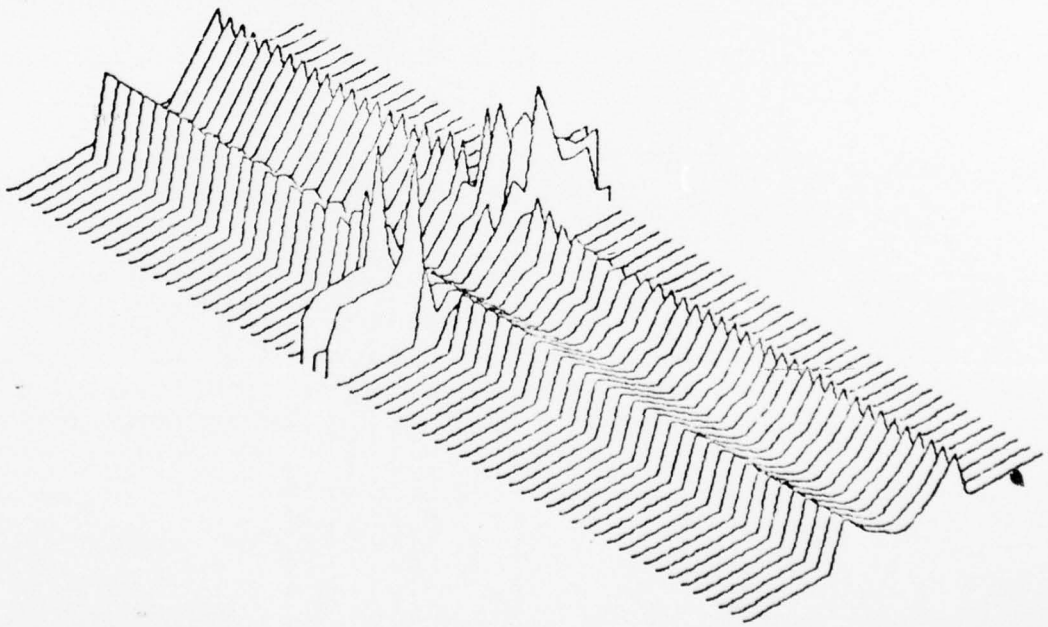
TOTAL VELOCITY

R= 500.

FRAME NO. 10.

Fig. 137: Total Velocity Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries; One Time Frame)





VORTICITY FUNCTION

R= 500.

FRAME NO. 10.

Fig. 138: Vorticity Function Plot for Idealized Renal Artery Section (Orthogonal Branching Arteries; One Time Frame)

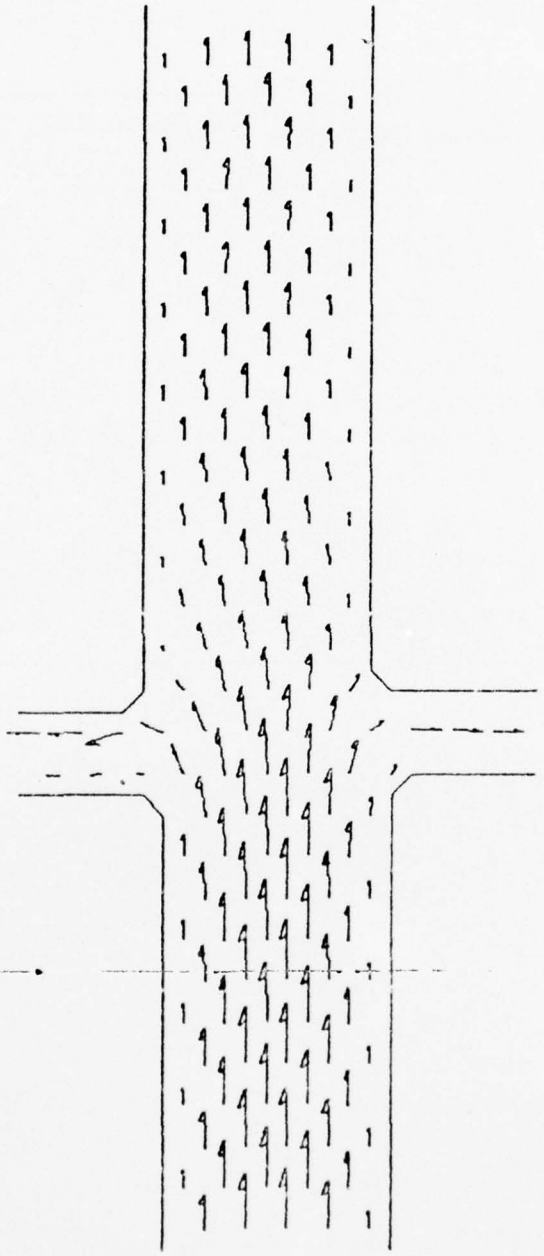


Fig. 139: Velocity Vector Plot for Idealized Renal Artery Section
(Orthogonal Branching Arteries; One Time Frame)

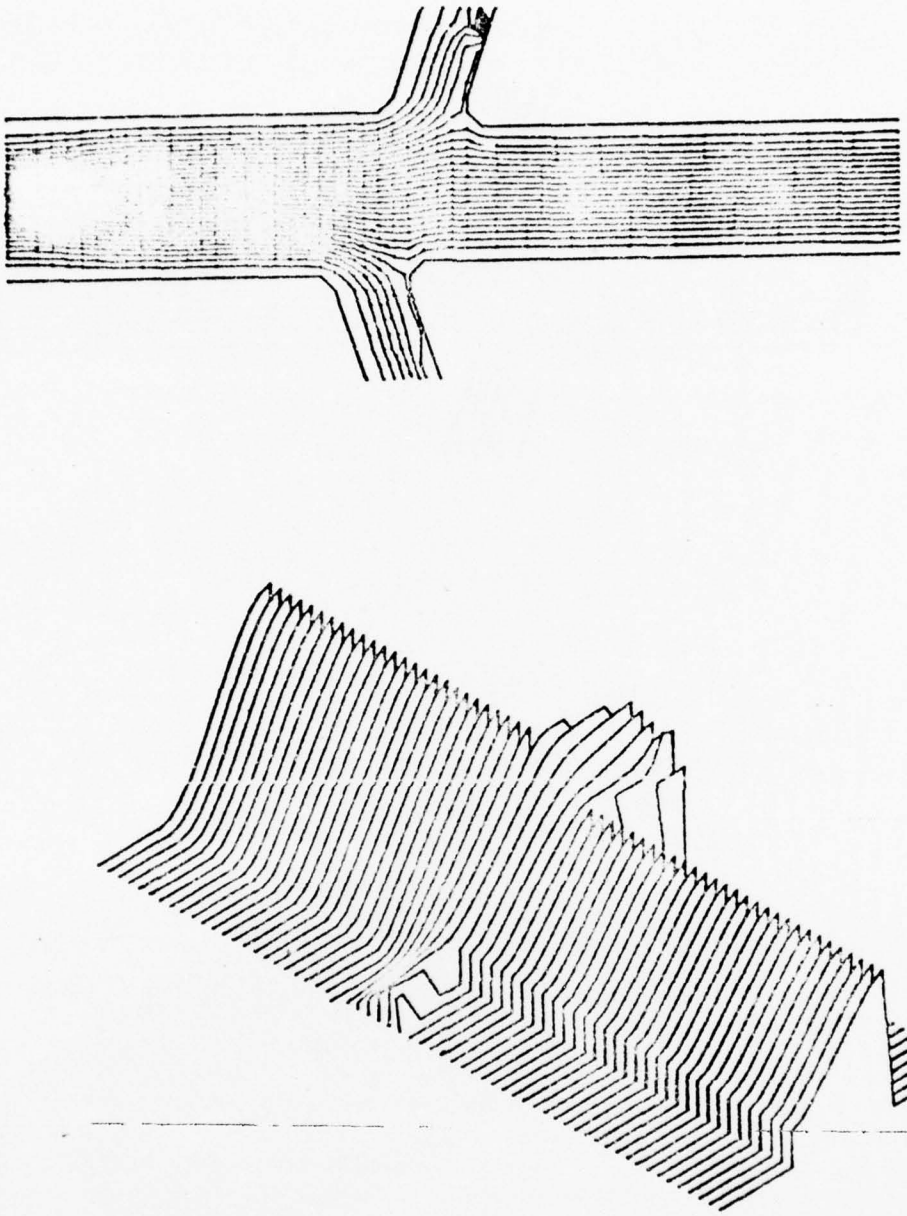


Fig. 140: Stream Function Plot for Non-Orthogonal Idealized Arterial Branching (Both Branches Face Downstream) At the Renal Section. One Time Frame.

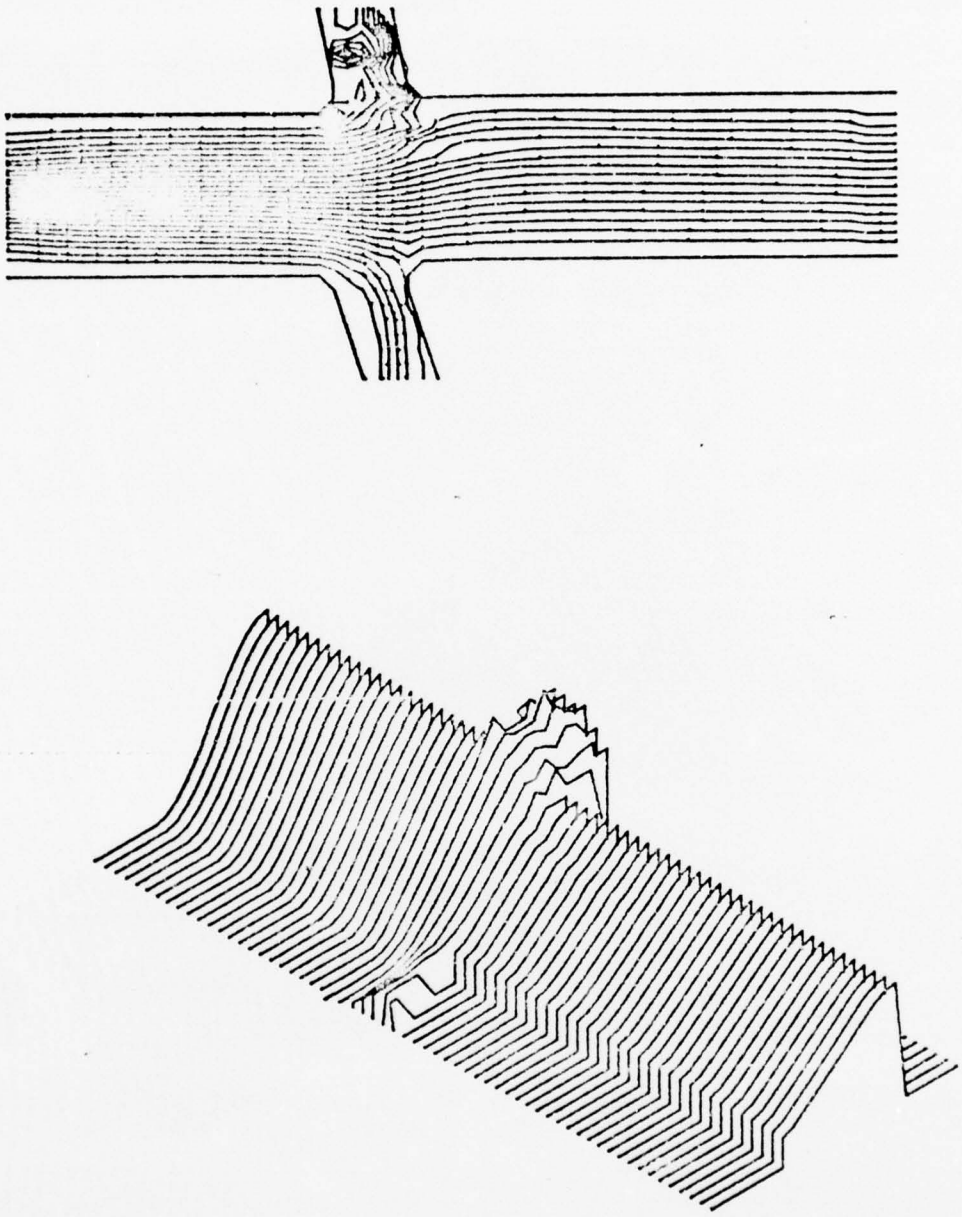
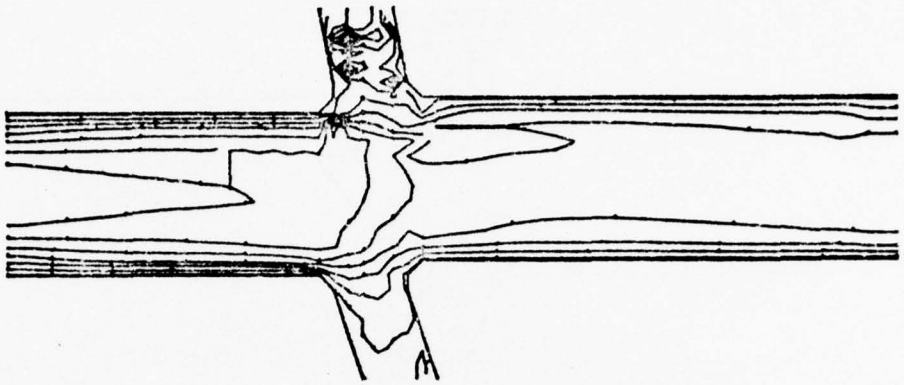
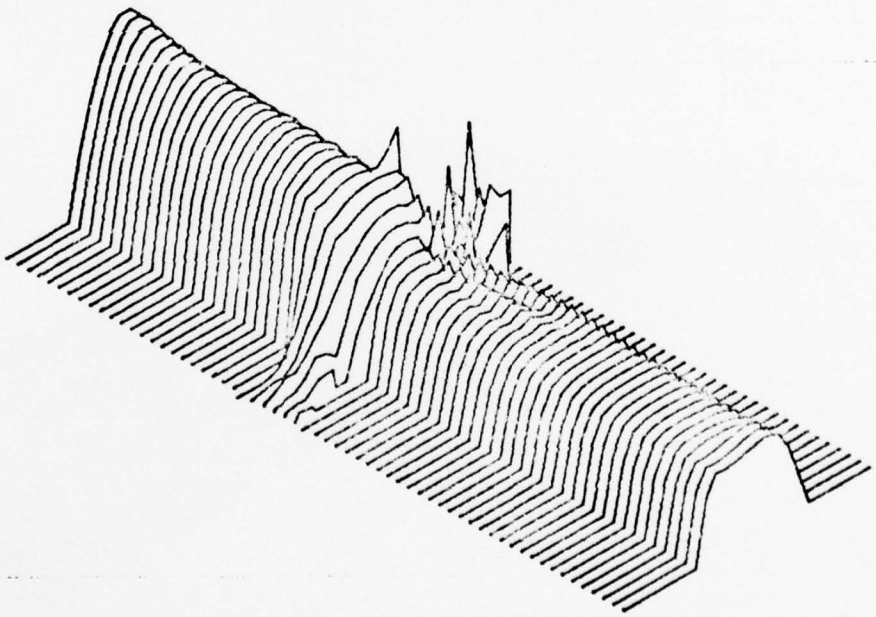


Fig. 141: Stream Function Plot for Branching Arteries at Renal Section (One Branch Faces Upstream and the Other Downstream) at the Renal Section. One Time Frame.





U VELOCITY

R= 1000.

FRAME NO. 10.

Fig. 142: Horizontal Velocity Function Plot for Branching Arteries at Renal Section (opposite Facing Branch Arteries). One Time Frame.

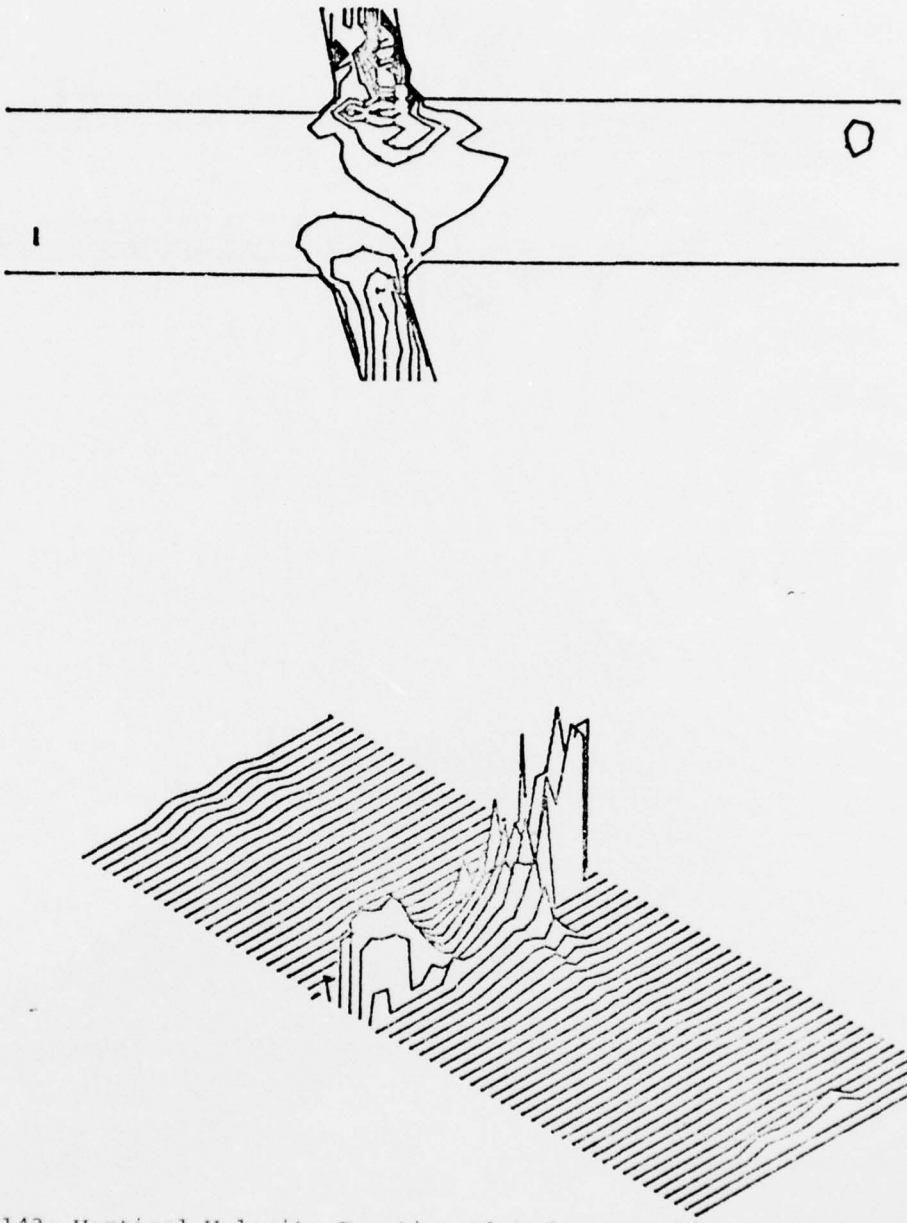
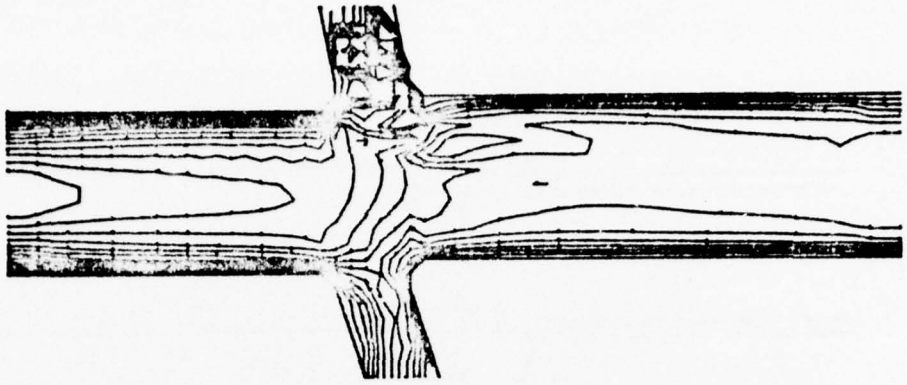


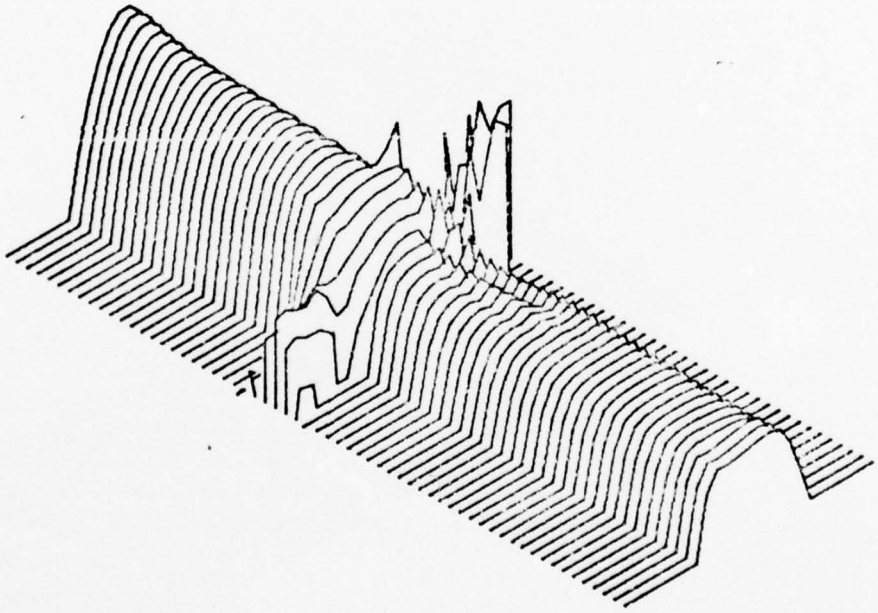
Fig. 143: Vertical Velocity Function Plot for Branching Artereis at Renal Section (One Branch Artery Faces Downstream and One Faces Upstream). One Time Frame.

V VELOCITY

R= 1000.

FRAME NO. 10.



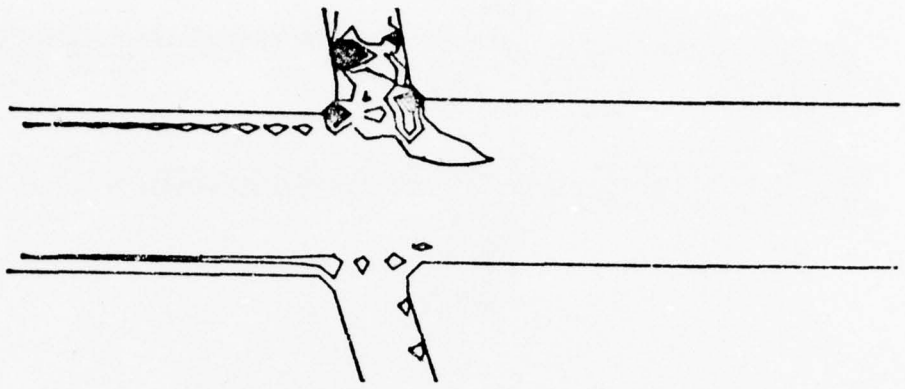


TOTAL VELOCITY

R= 1000.

FRAME NO. 10.

Fig. 144: Total Velocity Function Plot for Branching Arteries at Renal Section (One Branch Artery Faces Upstream and One Faces Dowstream). One Time Frame.



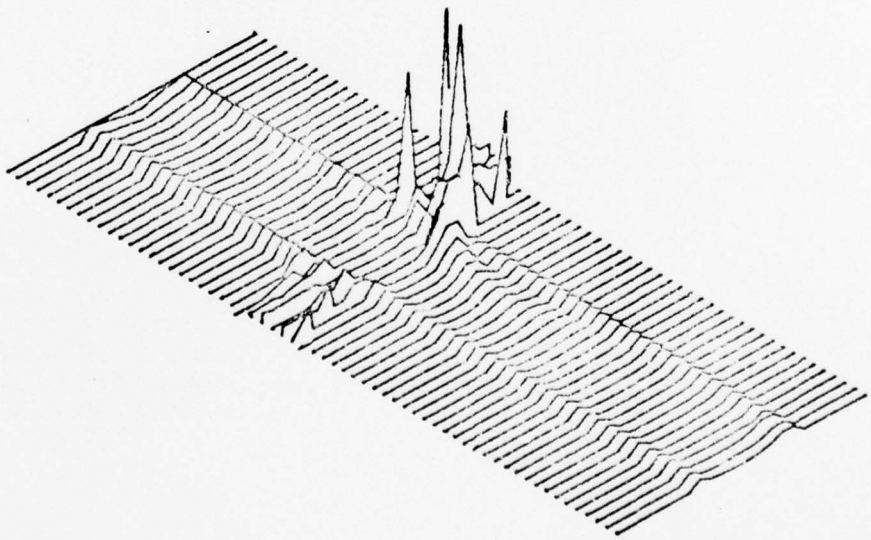
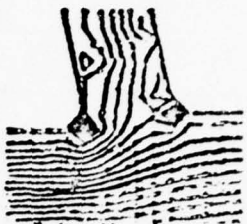


Fig. 145: Vorticity Function Plot for Branching Arteries at Renal Section (One Branch Artery Faces Upstream and the Other Faces Downstream). One Time Frame.

Reynolds Number = 20

Reynolds Number = 30



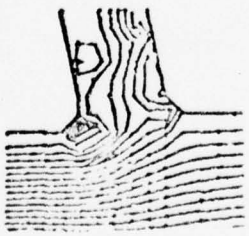
Reynolds Number = 40

Reynolds Number = 60

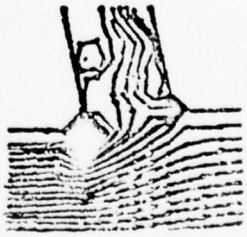
Reynolds Number = 70

Reynolds Number = 80

Reynolds Number = 90



Reynolds Number = 110



Reynolds Number = 120

Reynolds Number = 130

Reynolds Number = 140

Fig. 146: Montage of Portion of Renal Section with Side Artery Facing Upstream. At Different Velocities, Showing Vortex Development.

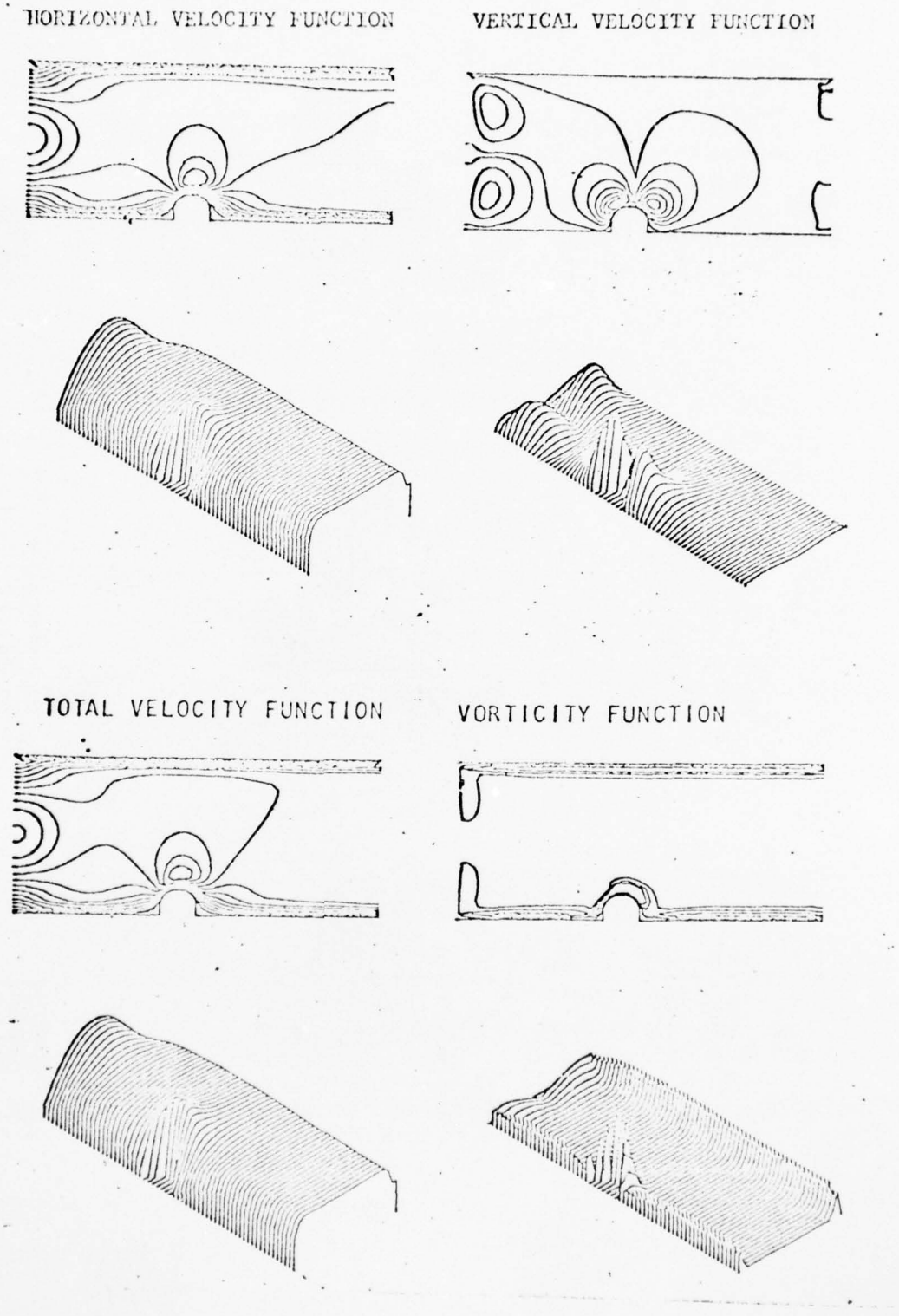
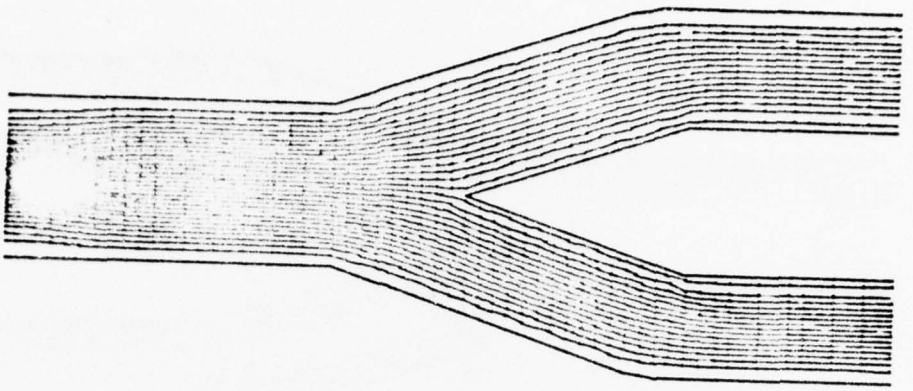
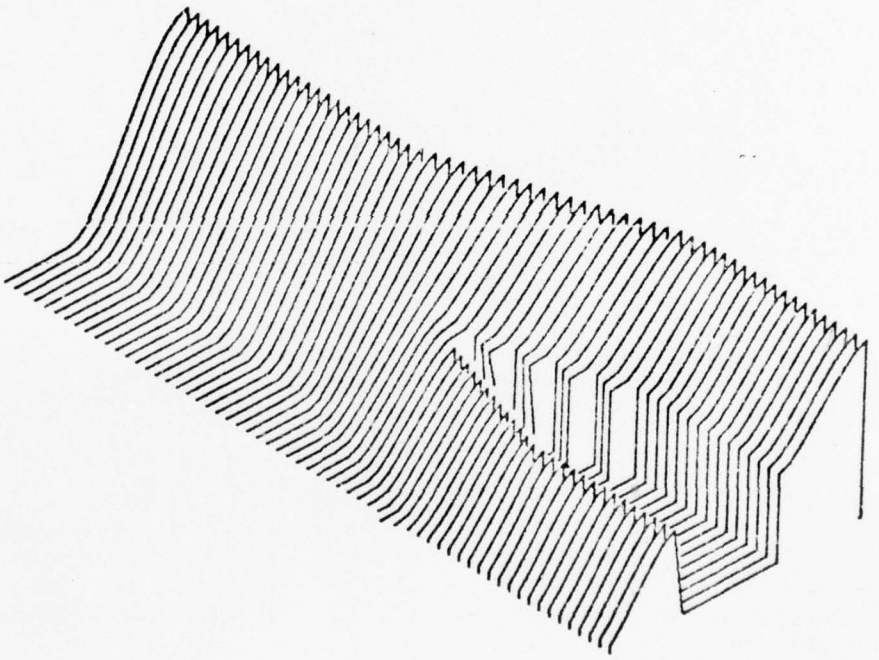


Fig. 147: A Montage That Shows Various Computer Devised Flow Functions For an Idealized Atherosclerotic Plaque.





STREAM FUNCTION

R= 150.

FRAME NO. 15.

Fig. 148: Stream Function Plot for Abdominal Aortic Bifurcation (One Time Frame)

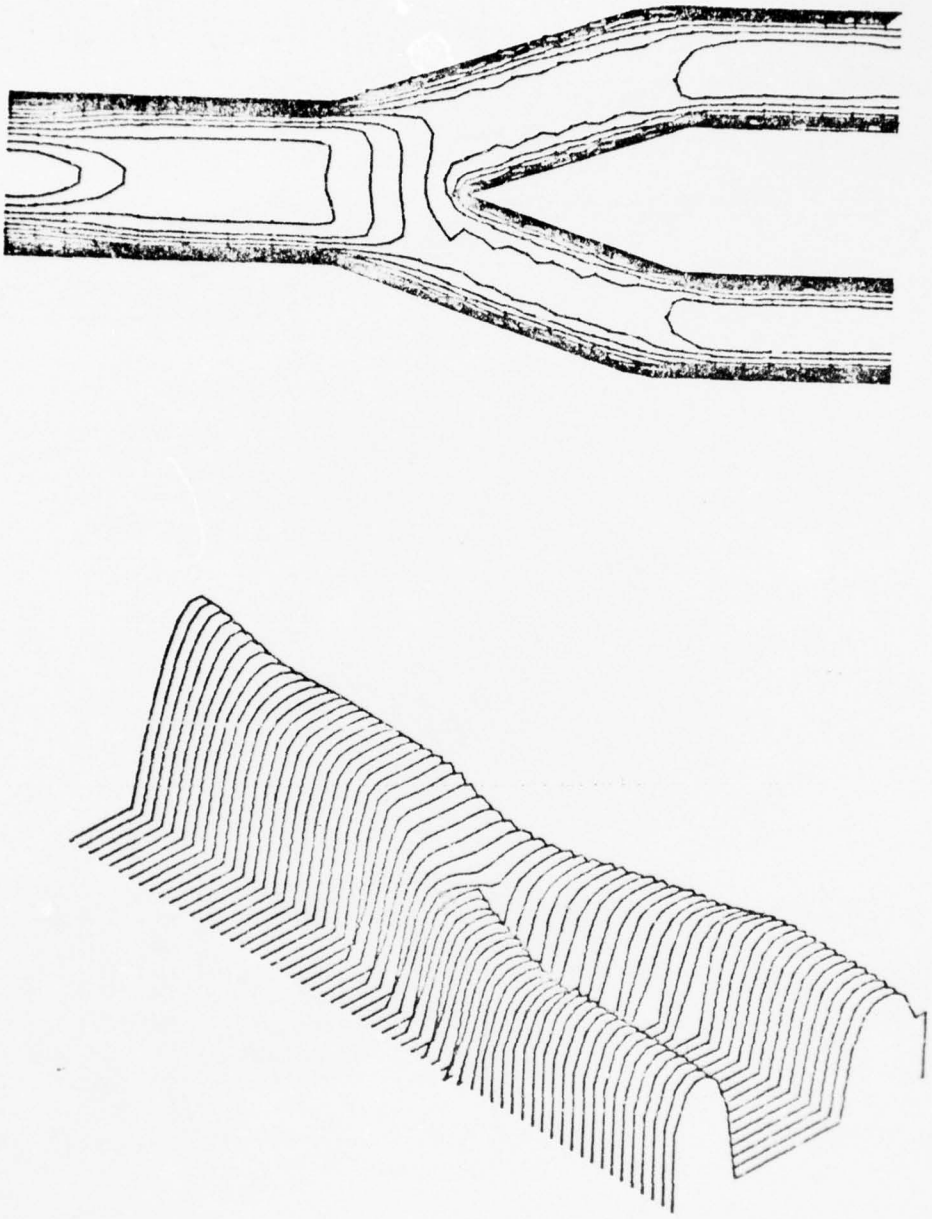
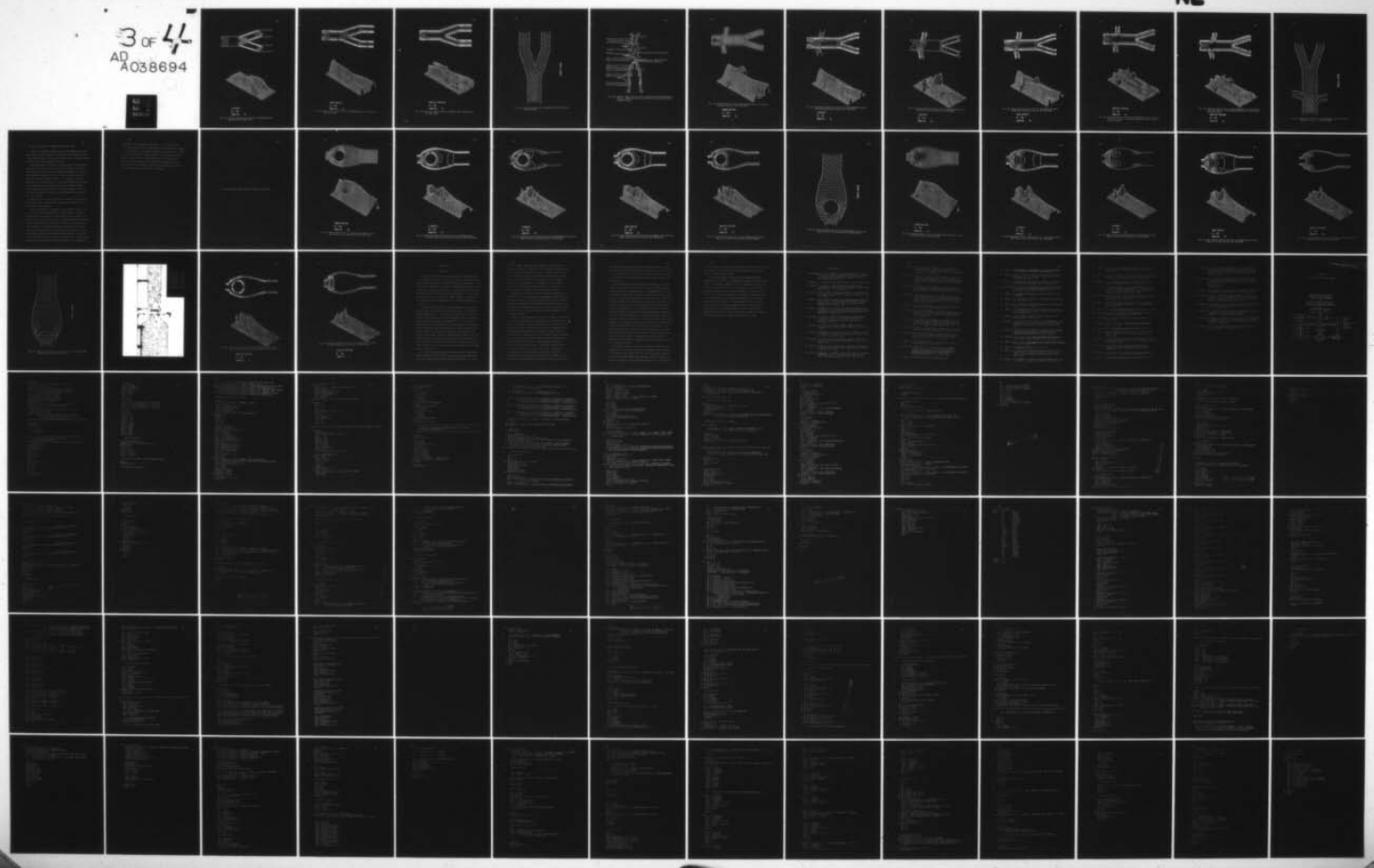


Fig. 149: Horizontal Velocity Function Plot for Abdominal Aortic Bifurcation (One Time Frame)

AD-A038 694 UTAH UNIV SALT LAKE CITY DEPT OF COMPUTER SCIENCE F/G 6/5
AN APPLICATION OF COMPUTER GRAPHICS: TWO CONCURRENT INVESTIGATI--ETC(U)
NOV 71 H GREENFIELD, R DEBRY F30602-70-C-0300
UNCLASSIFIED UTEC-CSC-71-115 NL

3 OF 4
AD A038694



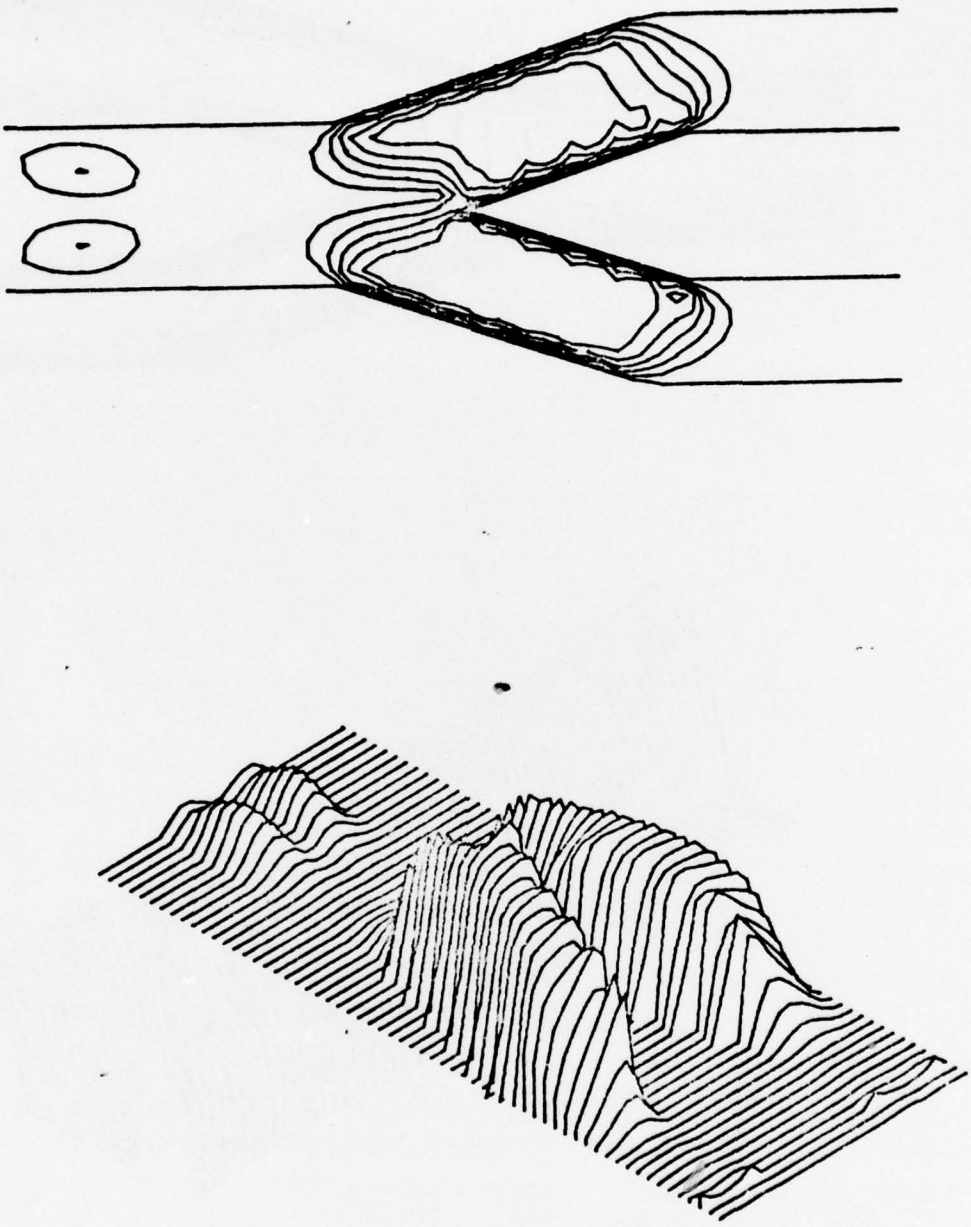


Fig. 150: Vertical Velocity Function Plot for Abdominal Aortic Bifurcation (One Time Frame).

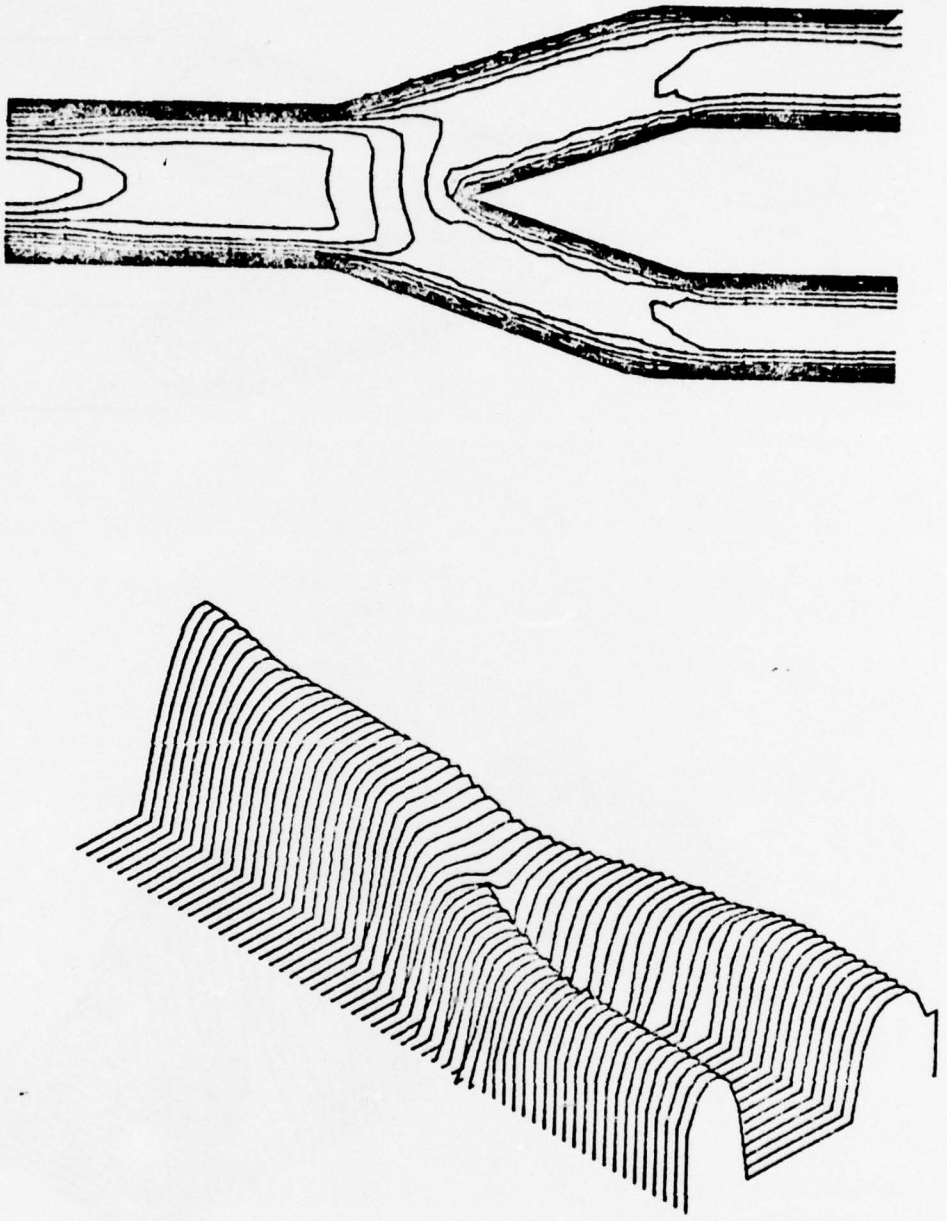
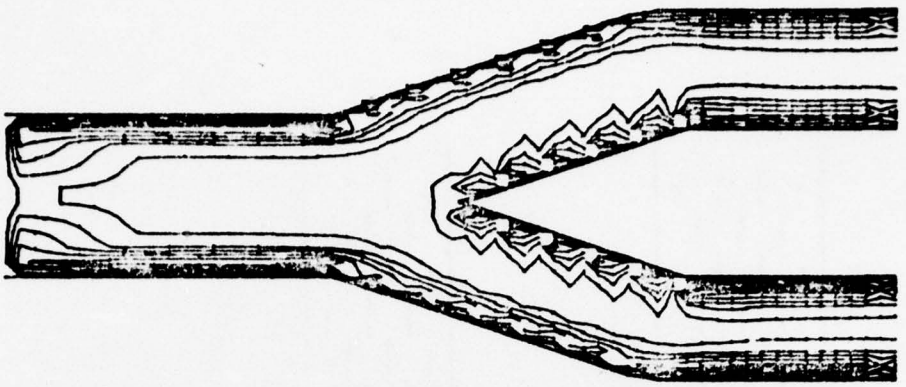
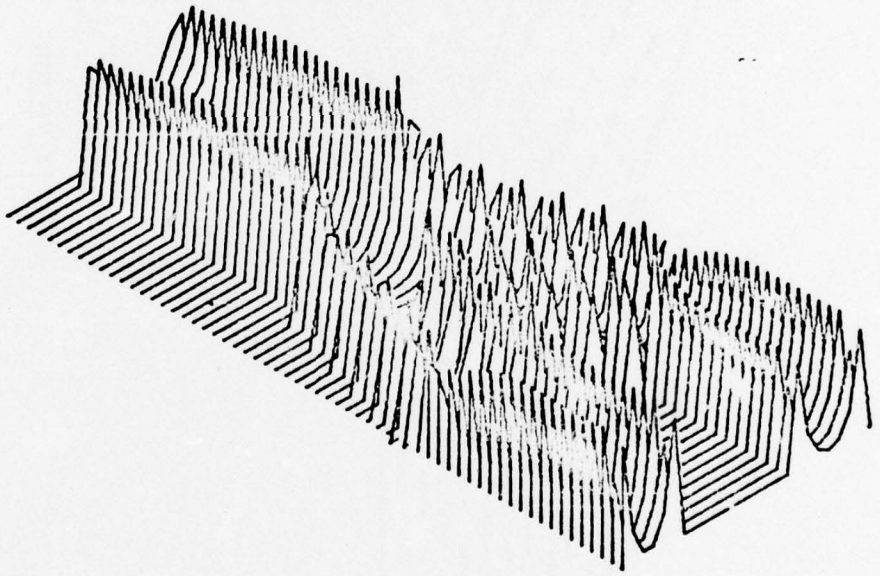


Fig. 151: Total Velocity Function Plot for Abdominal Aortic Bifurcation (One Time Frame)





VORTICITY FUNCTION

R= 150.

FRAME NO. 15.

Fig. 152: Vorticity Function Plot for Abdominal Aortic Bifurcation (One Time Frame)

Fig 6

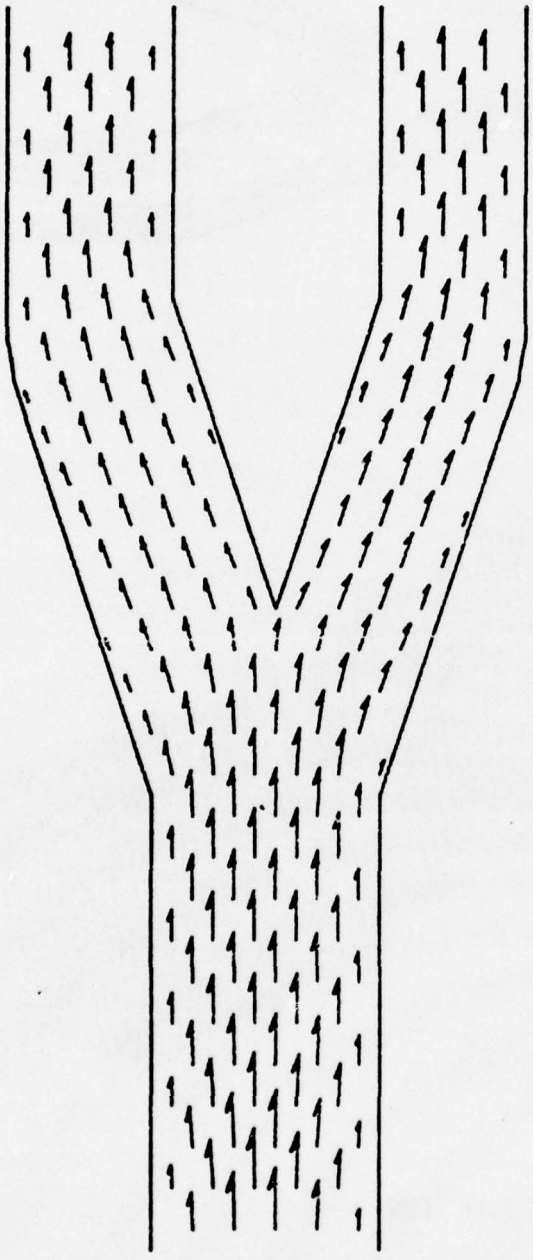


Fig. 153: Velocity Vector Plot for Abdominal Aortic Bifurcation
(One Time Frame)

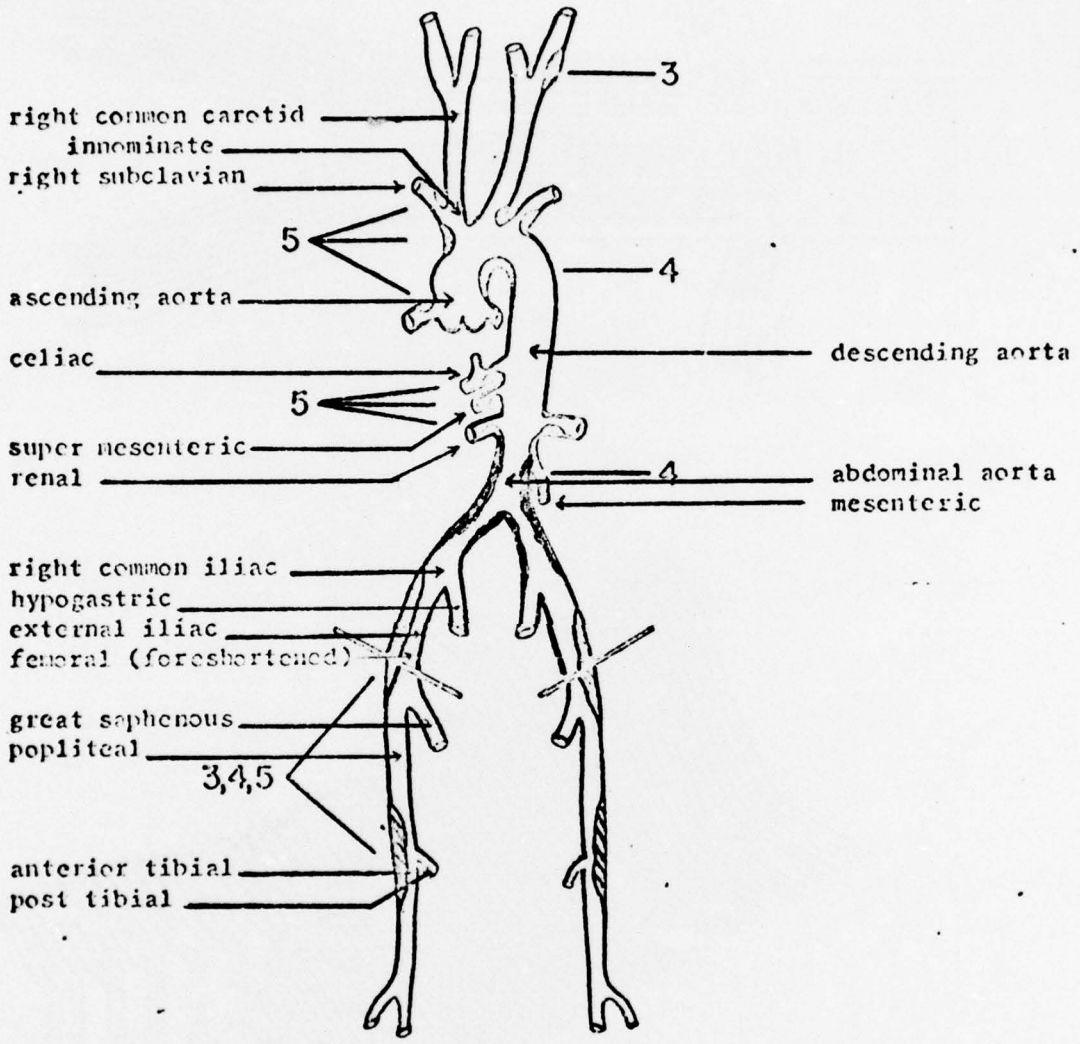


Fig. 154: Arterial "Tree" Showing Sites of Predilection for Atherosclerotic Plaques. (Note Dark Sections Along Walls at Mesenteric Section) From Wesolowski[23].

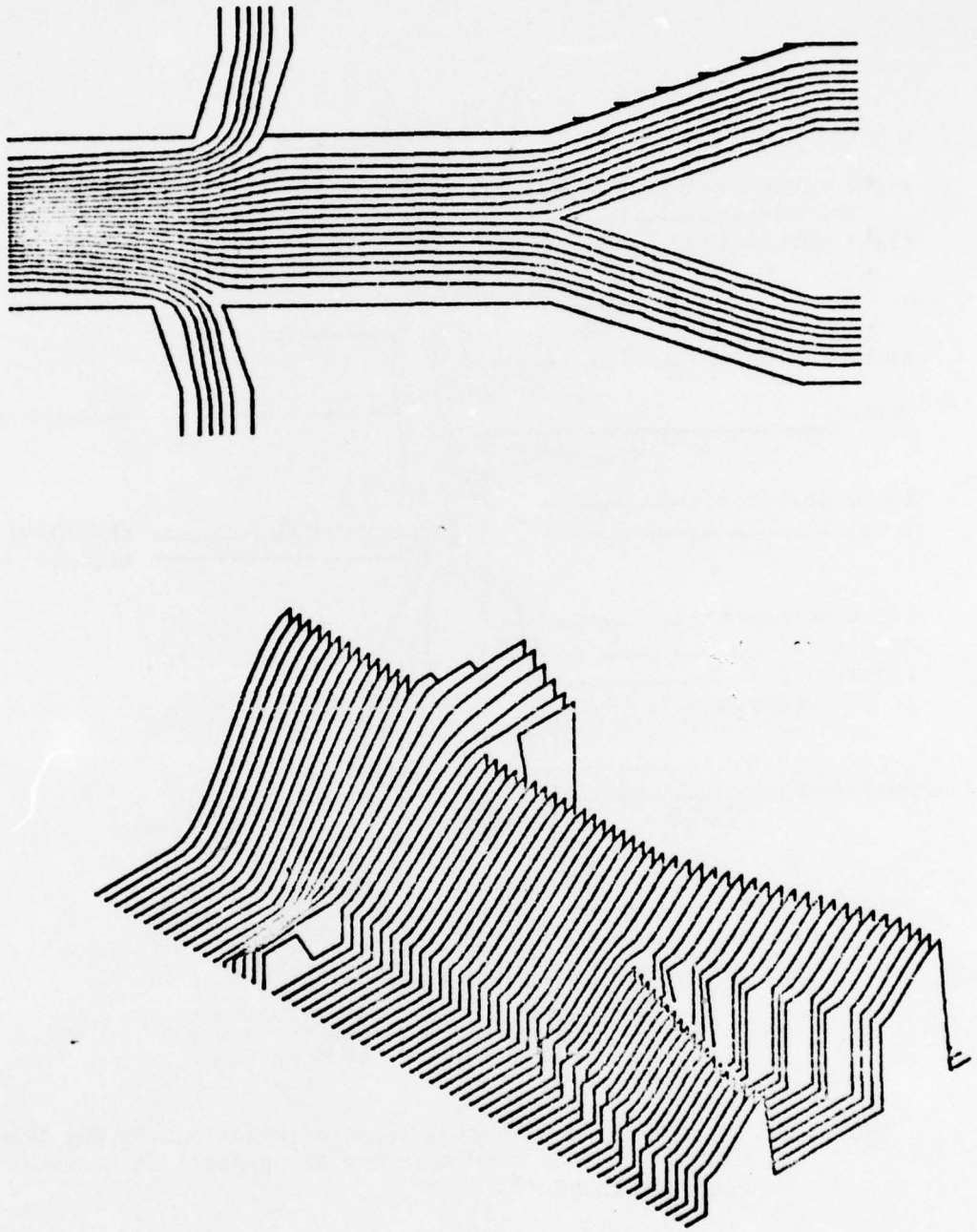


Fig. 155: Stream Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; One Time Frame.

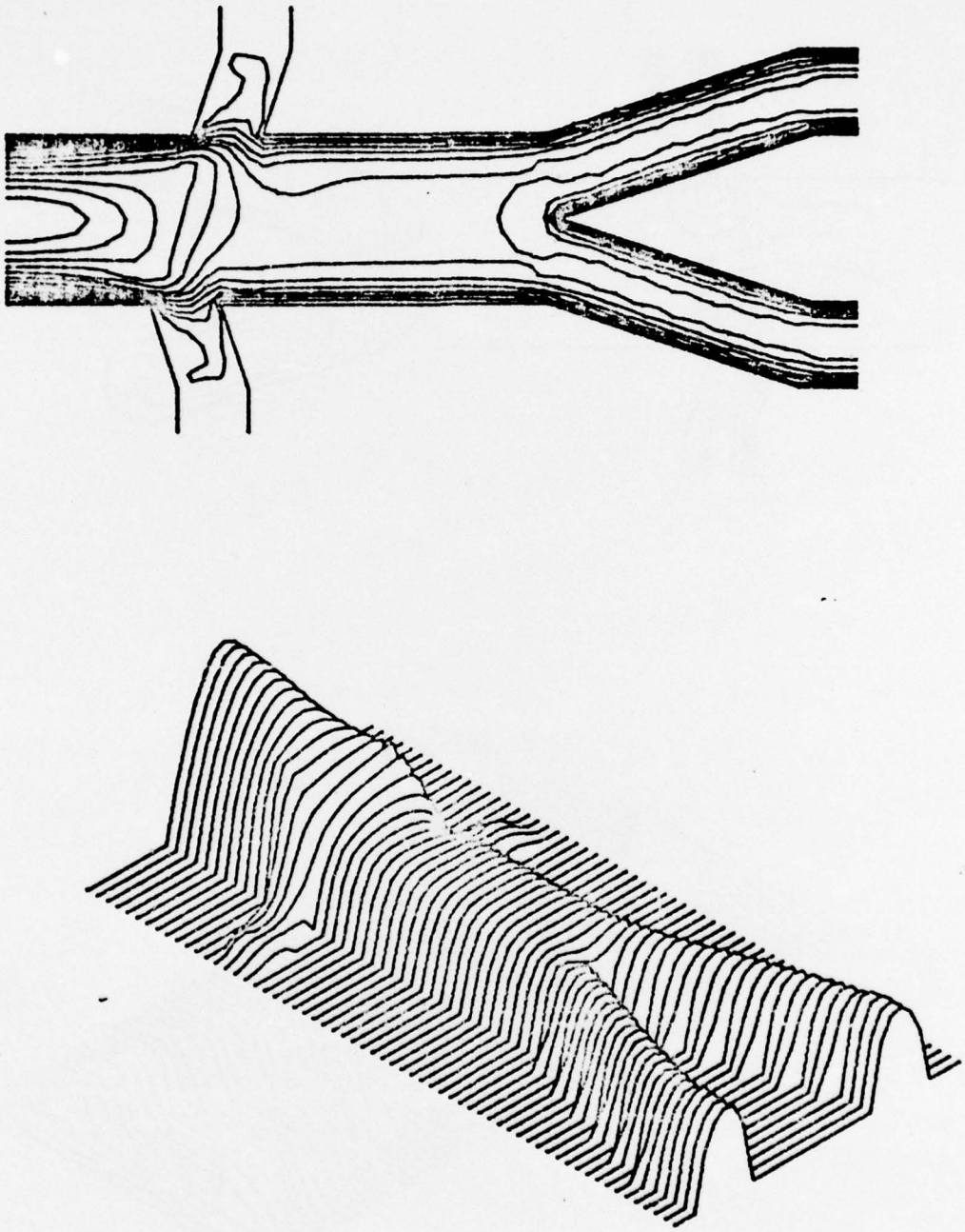
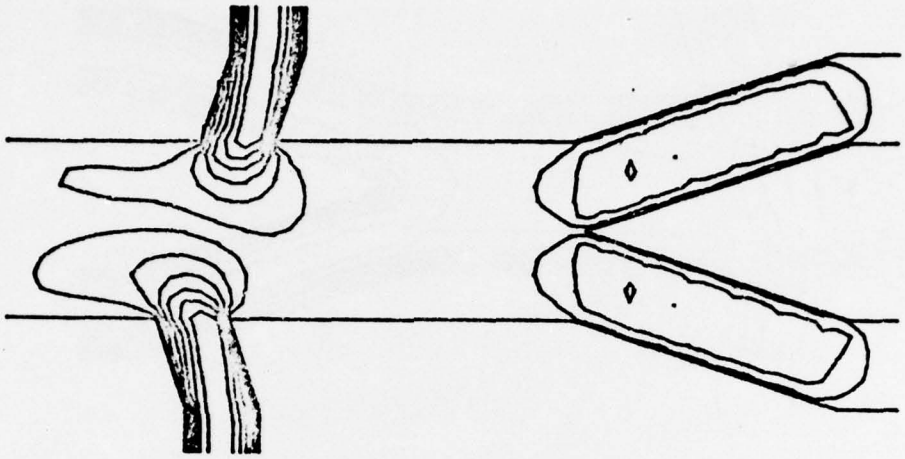


Fig. 156: Horizontal Velocity Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; One Time Frame.



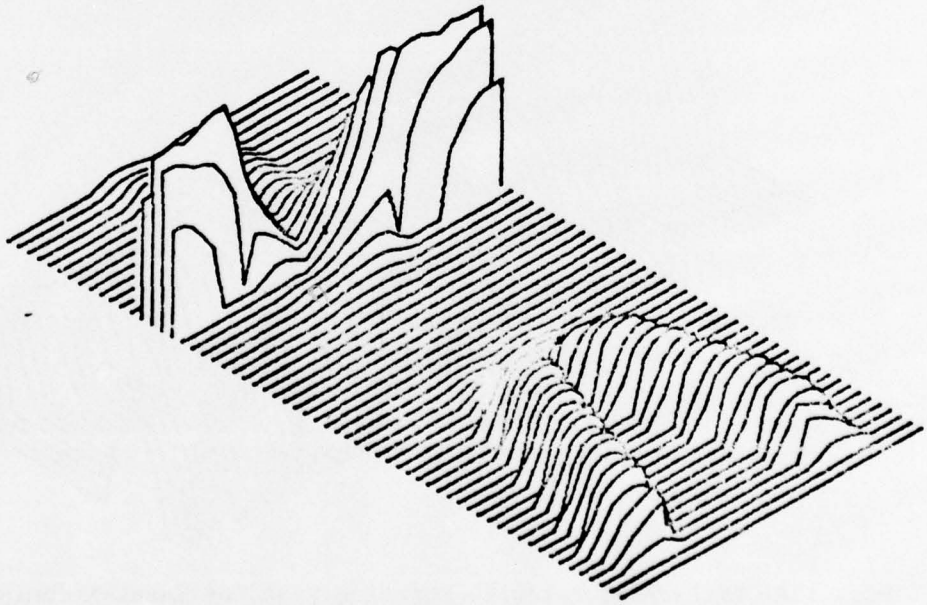


Fig. 157: Vertical Velocity Function Plot for Renal-Abdominal Aortic Bifurcation Section Combined; One Time Frame.

V VELOCITY

R= 150.

FRAME NO. 15.

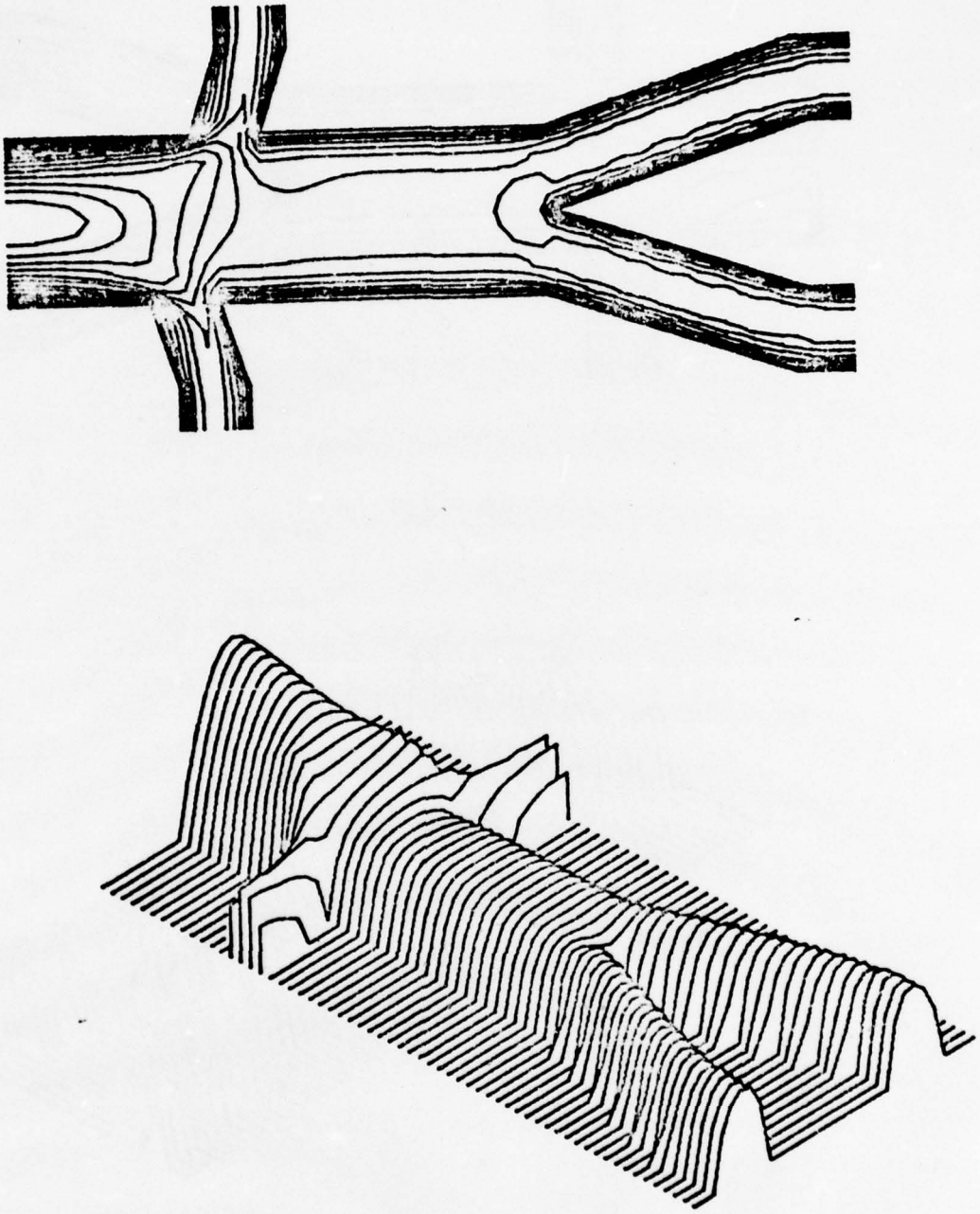


Fig. 158: Total Velocity Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; One Time Frame.

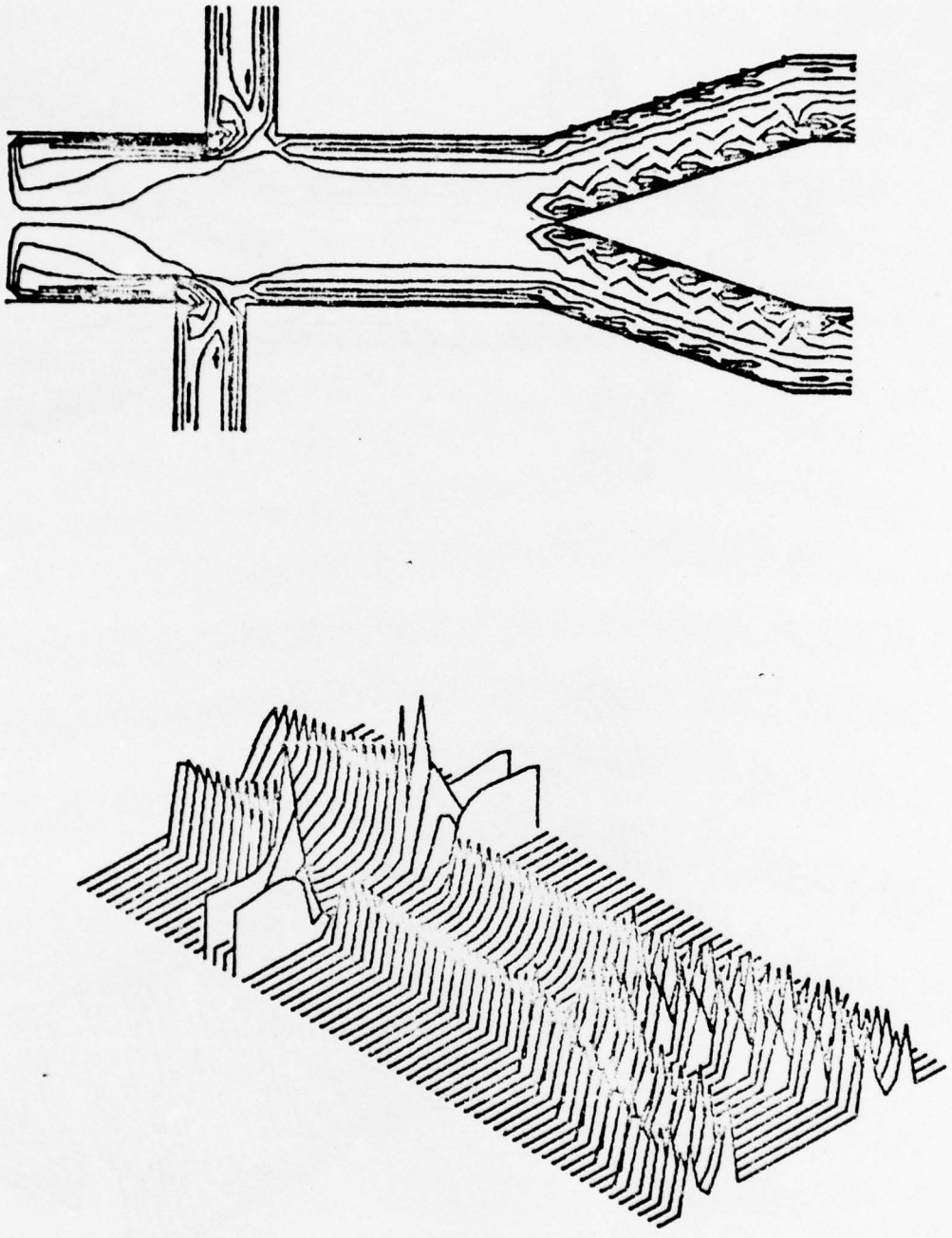


Fig. 159: Vorticity Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; Orthogonal Branching at Renal Position. (One Time Frame).

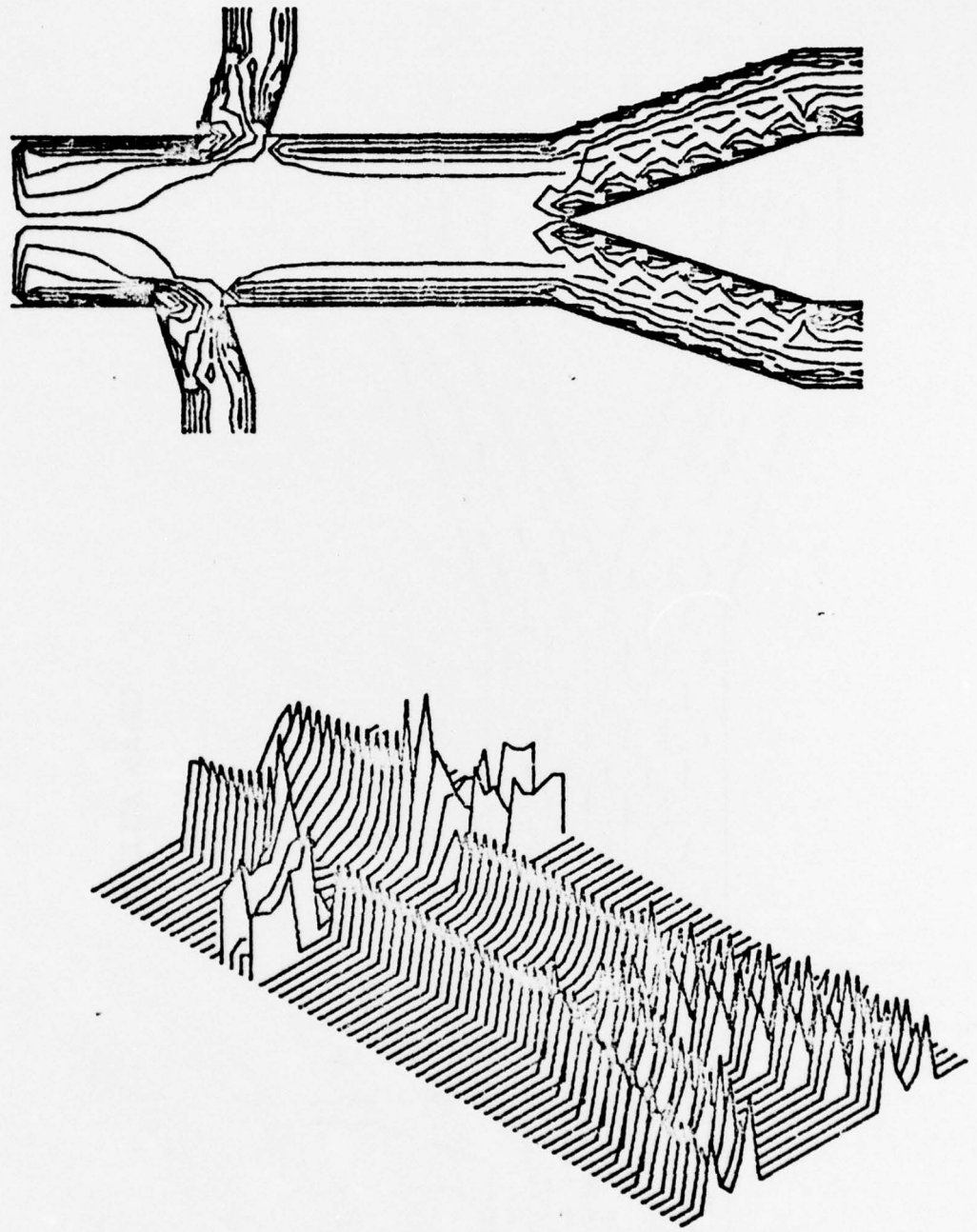


Fig. 160: Vorticity Function Plot for Renal-Abdominal Aortic Bifurcation Combination Section; Non-Orthogonal Branching at Renal Section (One Time Frame).

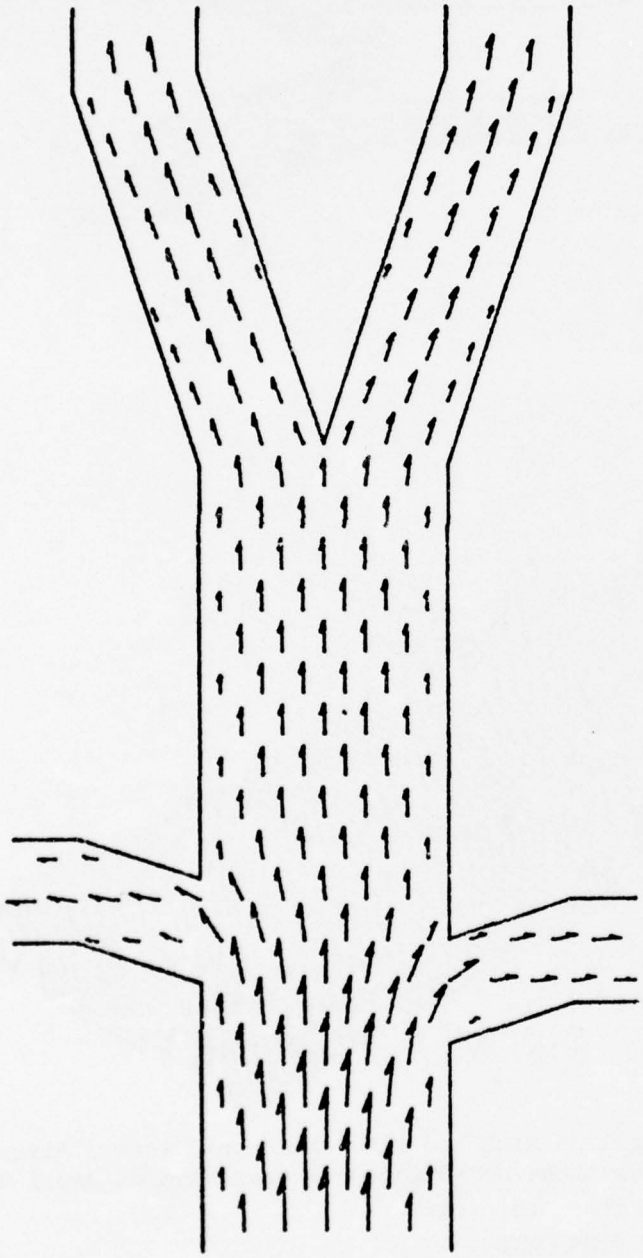


Fig. 161: Velocity Vector Plot for Renal-Abdominal Aortic Bifurcation Combination Section. (One Time Frame)

C. Discussion of Concern to Prosthetic Heart Valve Study.

Figures 162 through 167 present one time frame examples of the computer graphic results as applied to an ideal ball type central occlusion artificial heart valve. In turn, figures 168 through 173 display equivalent plots for a disc type artificial heart valve.

Remembering that the computer is an inert brute with non-diagnostic capabilities in the clinical sense, perhaps the advocates of the disc valve will be pleased to notice what qualitatively appears to be larger vortex formations distal to the ball in figure 166, than for the disc valve, when viewing the isometric result. In comparison, ball valve proponents, upon viewing figure 172, could perhaps smugly point to heavy vortex concentration at the edges of the discoid, possibly leading to quicker edge wear for this valve type. With such a thought in mind, the ball type also appears to show more evenly distributed vorticity motion about the occluder.

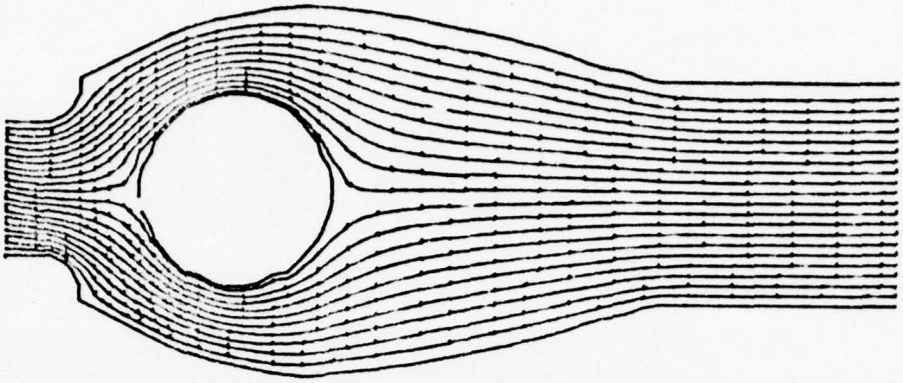
Figures 175 and 176 show the ball and disc valves in less than full open position as the computer studies them when the occluder is closer to the valve's sewing position.

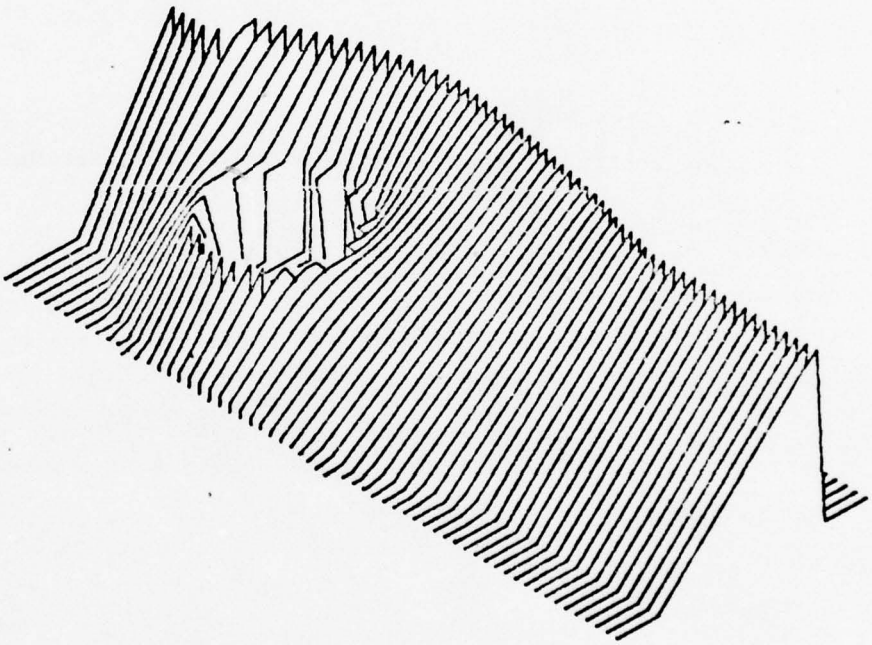
Again, qualitatively, the isometric views show higher peaks for vorticity which is to be expected when the shearing effects are greater between the artery wall and the occluder as this space is lessened. As one should also assume intuitively, the "computer experiment" shows heavier vortex concentration about both types of occluders as they move towards closure position. One can note the extent and activity of the vortex concentration, when the valve occluder for either valve type is not fully opened, when the observer realizes that figures 175 and 176 were formed at the second time frame after initiation of steady flow. In turn, figures 166 and 172 were formed at the fifteenth time frame. This section of the

investigation is continuing, as time permits, since closeness of the artery wall requires a finer computational mesh for the numerical analysis.

Most important is the comparison between figures 172 and 174. Figure 174 is a flow pattern as formed by Wieting [59] while photographing the motion of particles about a Kay-Shiley valve in a pulse duplication system. Wieting's experiments were performed independently at Baylor University and for preliminary comparison of vortex positioning with the present computer result, the product is gratifying.

D. Curved Boundary Computer Graphic Displays (Heart Valve).



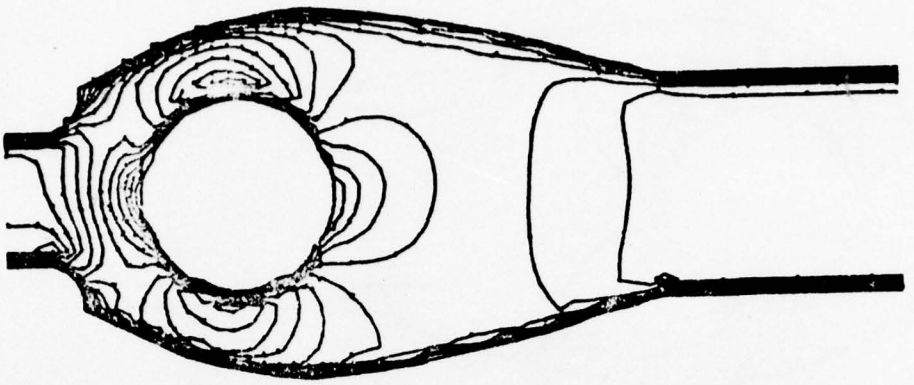


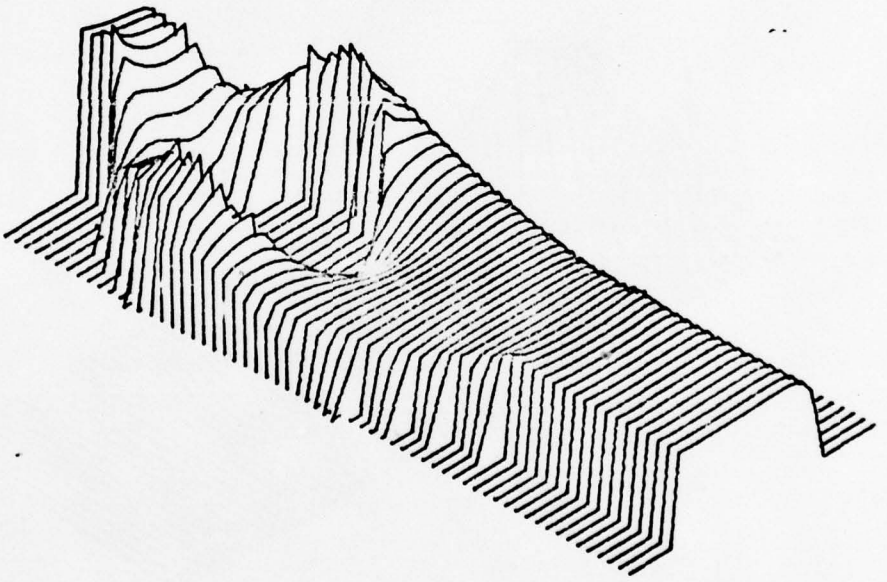
STREAM FUNCTION

R= 150.

FRAME NO. 15.

Fig. 162: Stream Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position (One Time Frame)



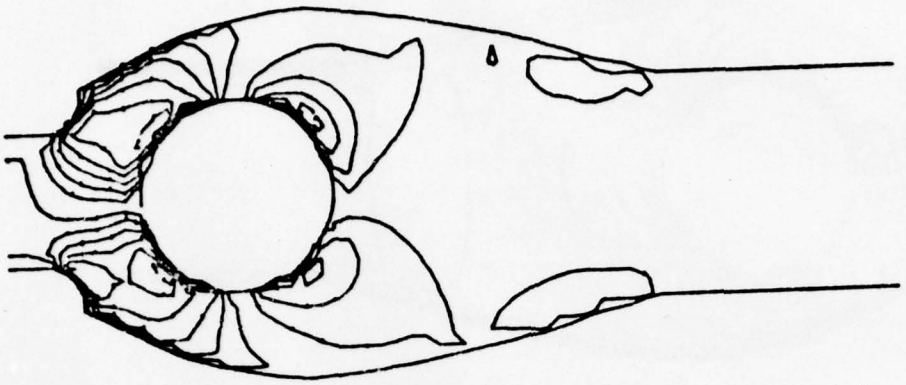


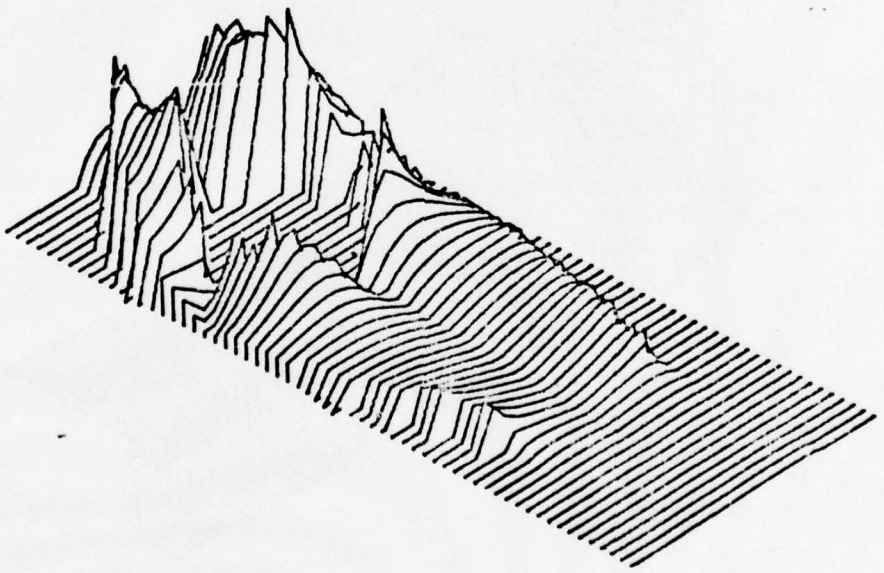
U VELOCITY

R= 150.

FRAME NO. 15.

Fig. 163: Horizontal Velocity Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position (One Time Frame)



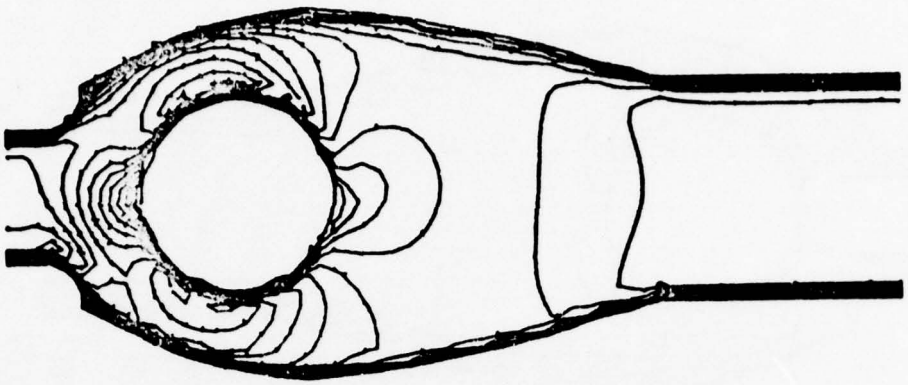


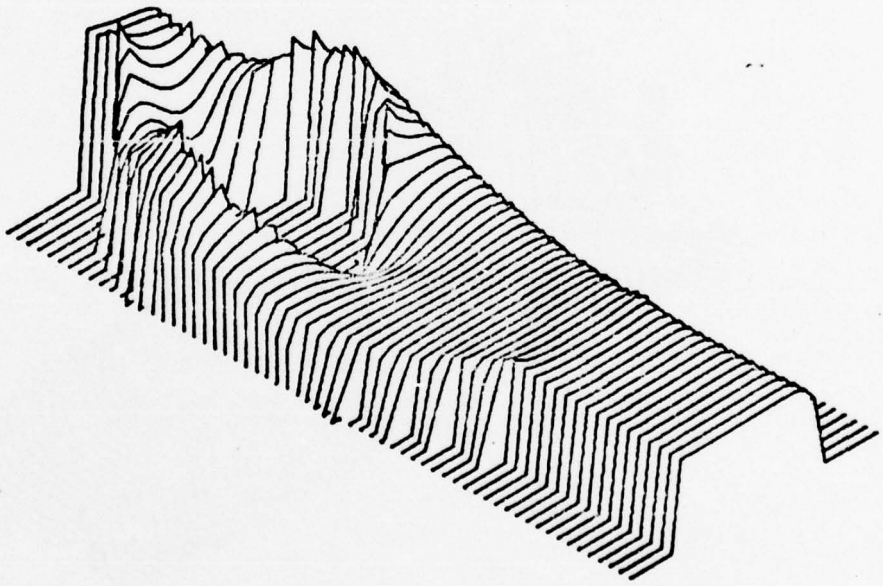
V VELOCITY

R= 150.

FRAME NO. 15.

Fig. 164: Vertical Velocity Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position (One Time Frame)



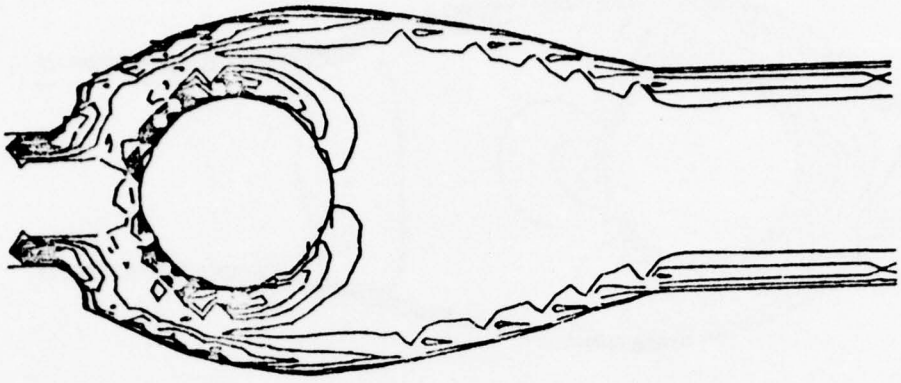


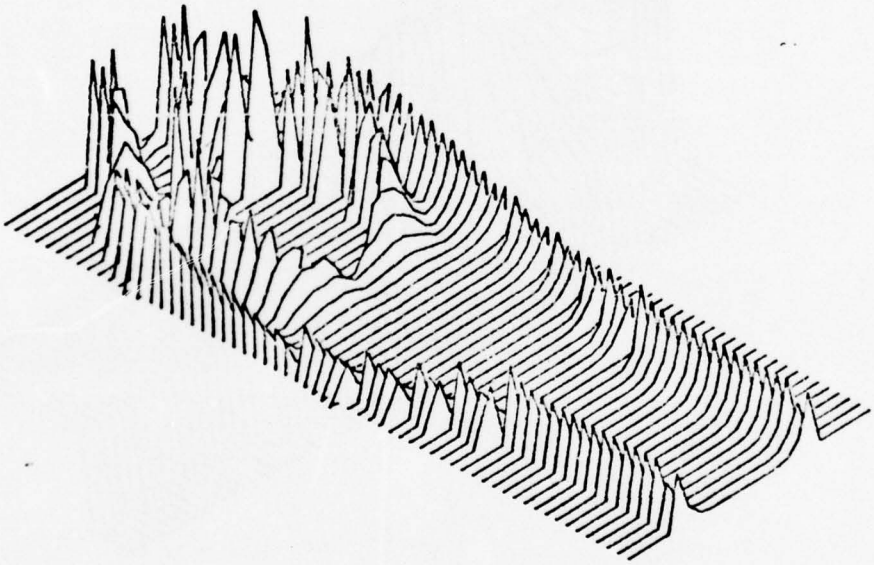
TOTAL VELOCITY

R= 150.

FRAME NO. 15.

Fig. 165: Total Velocity Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position (One Time Frame)





VORTICITY FUNCTION

R= 150.

FRAME NO. 15.

Fig. 166: Vorticity Function Plot for Starr-Edwards Type Artificial Heart Valve at Full Open Position (One Time Frame)

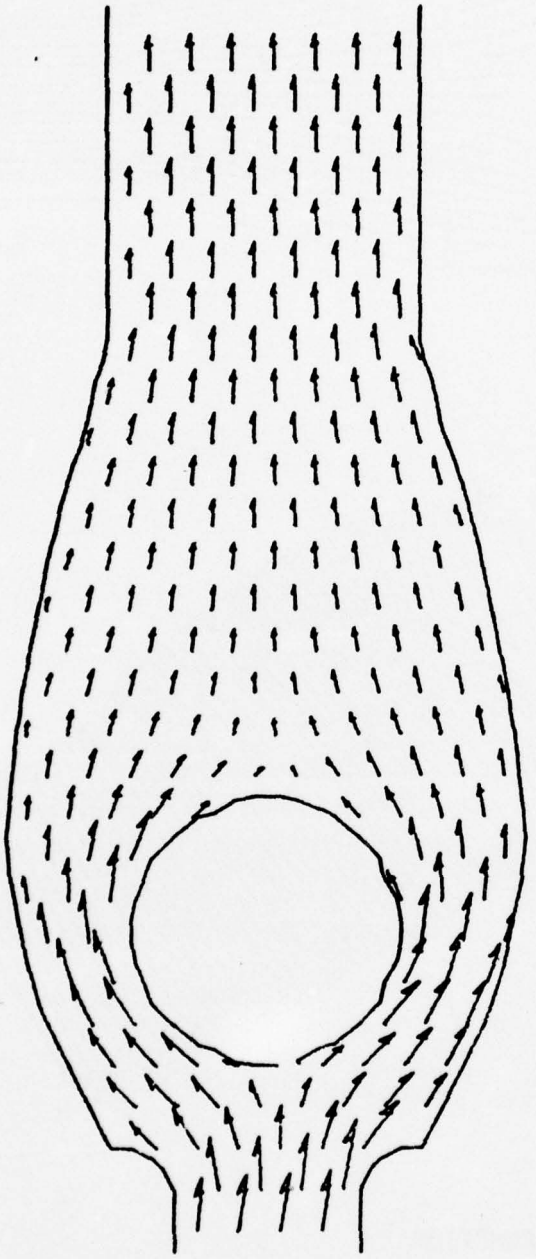


Fig. 167: Velocity Vector Plot for Starr-Edwards Heart Valve at Full Open Position at Full Open Position (One Time Frame).

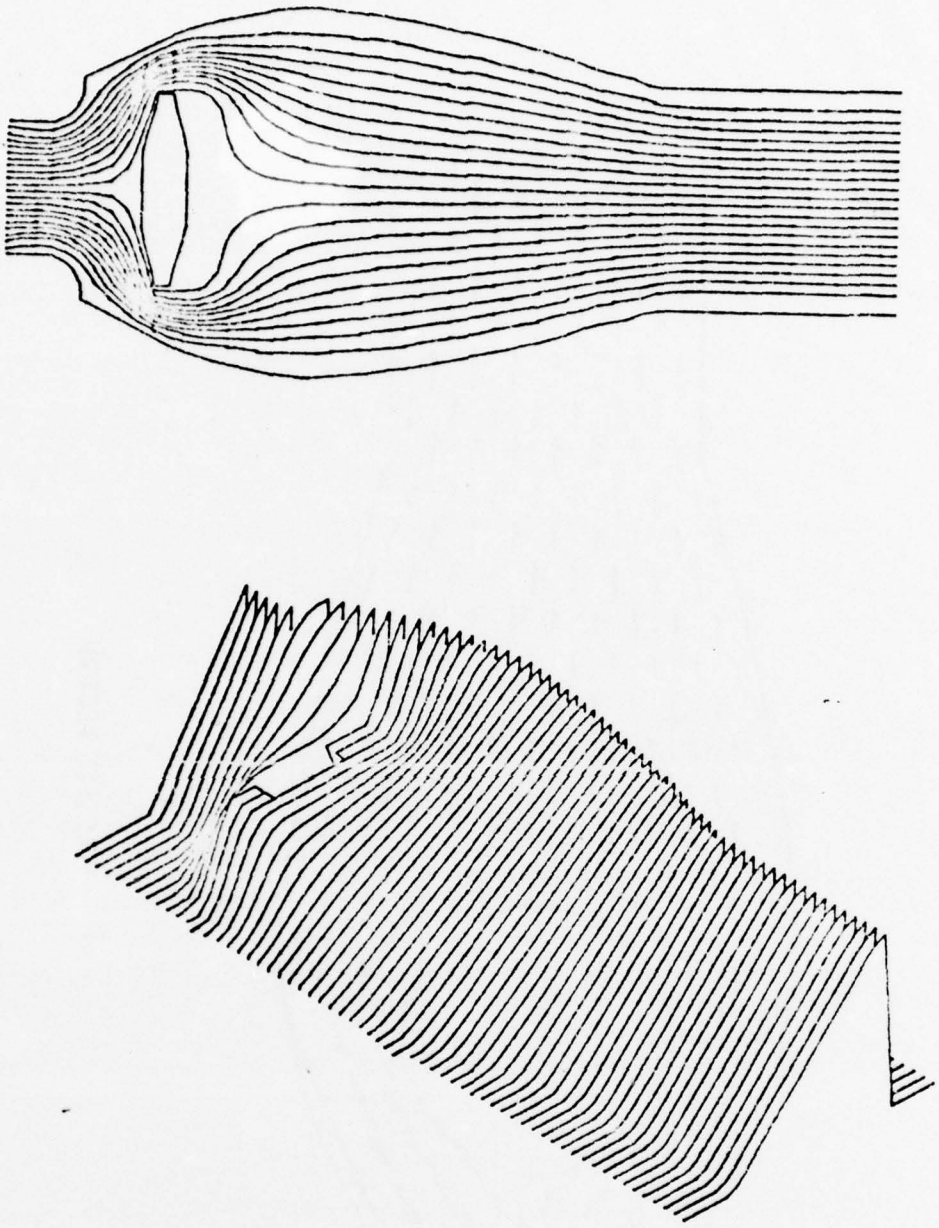


Fig. 168: Stream Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position (One Time Frame)

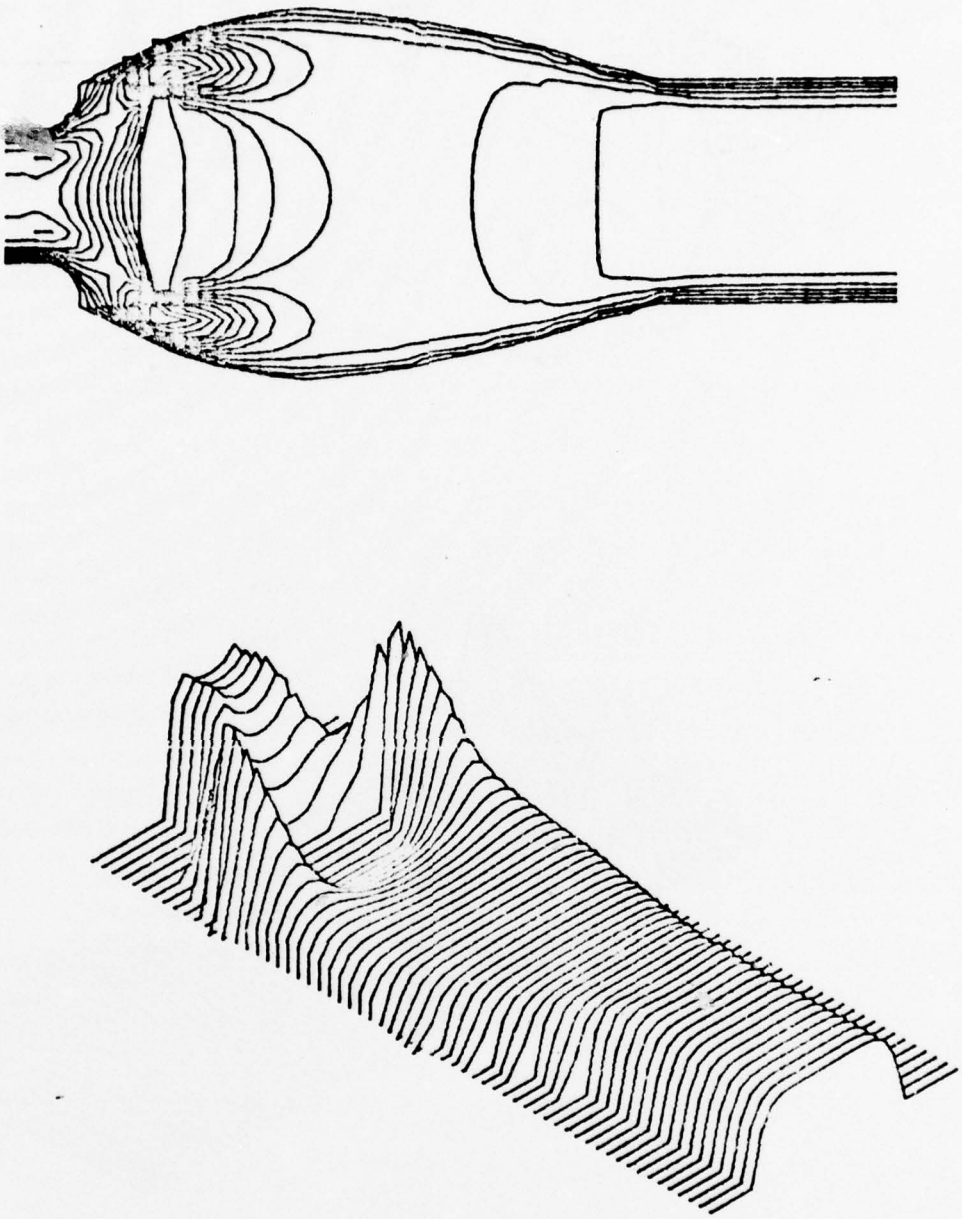
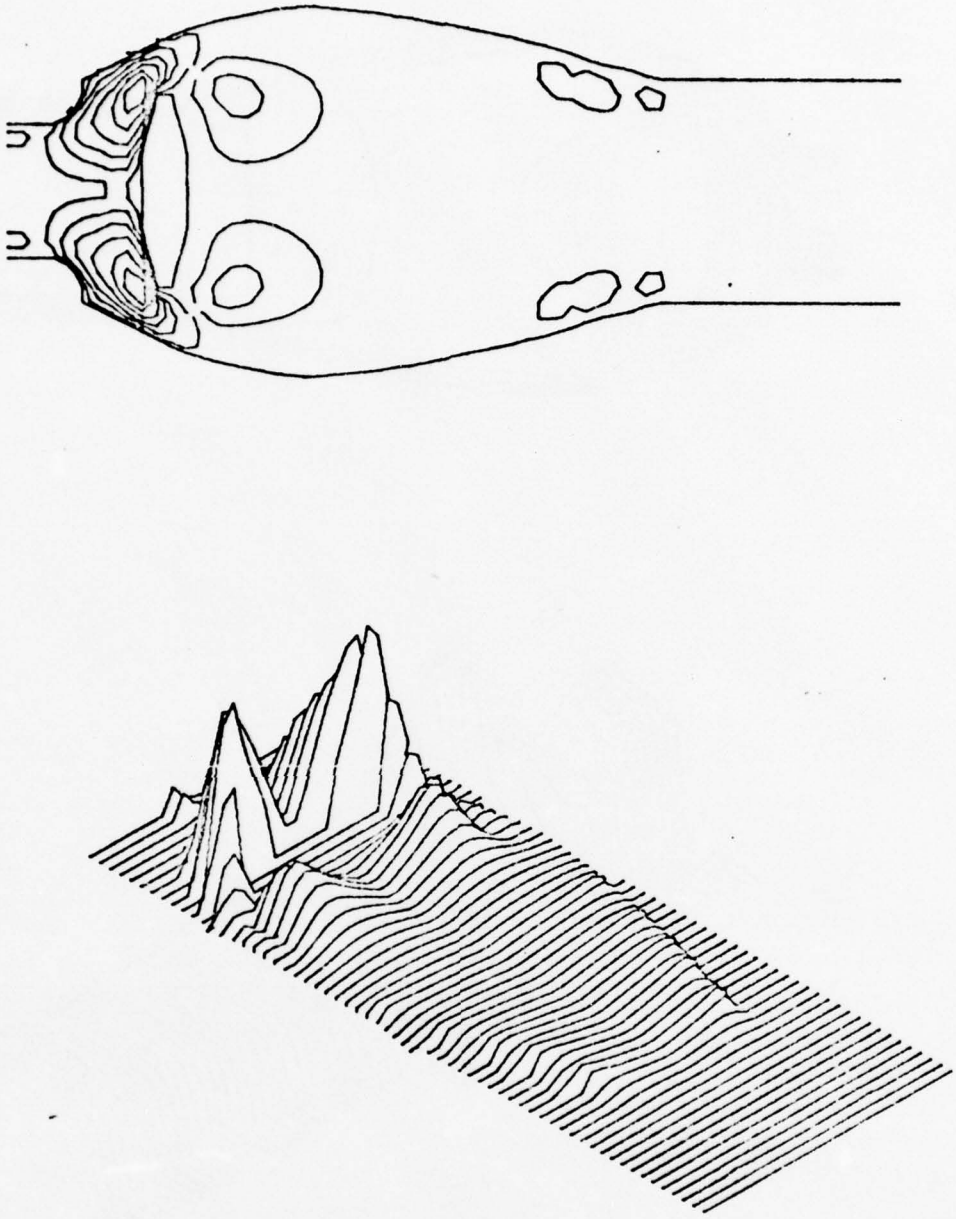


Fig. 169: Horizontal Velocity Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position (One Time Frame)

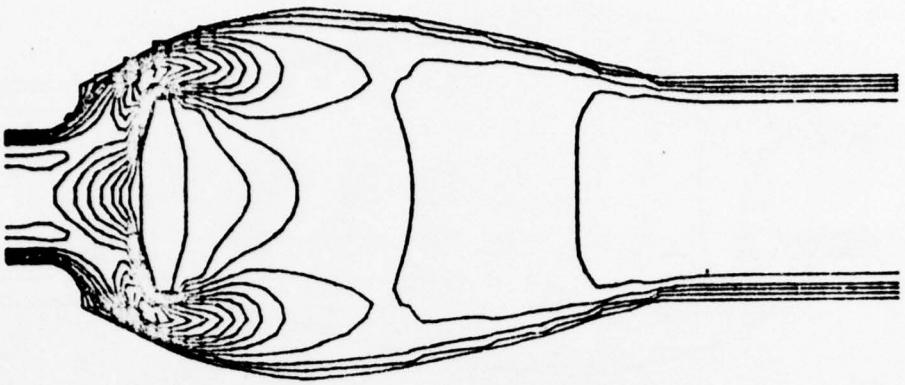


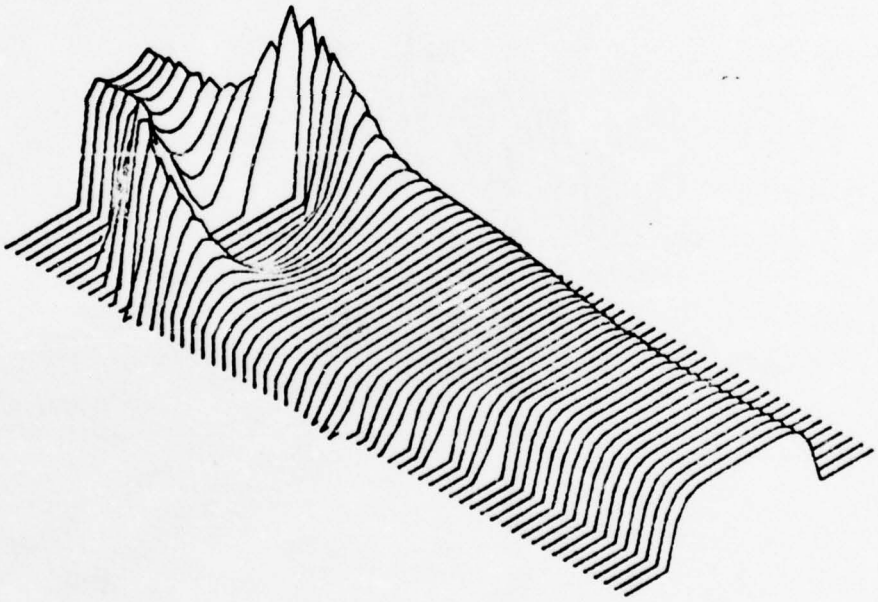
V VELOCITY

R= 150.

FRAME NO. 15.

Fig. 170: Vertical Velocity Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position (One Time Frame)



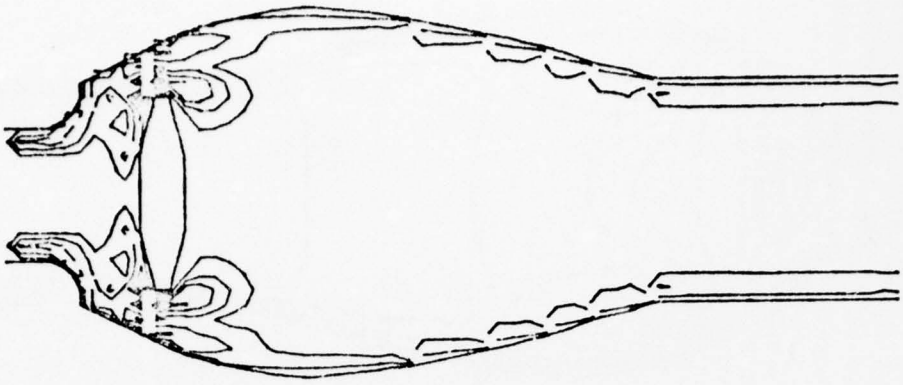


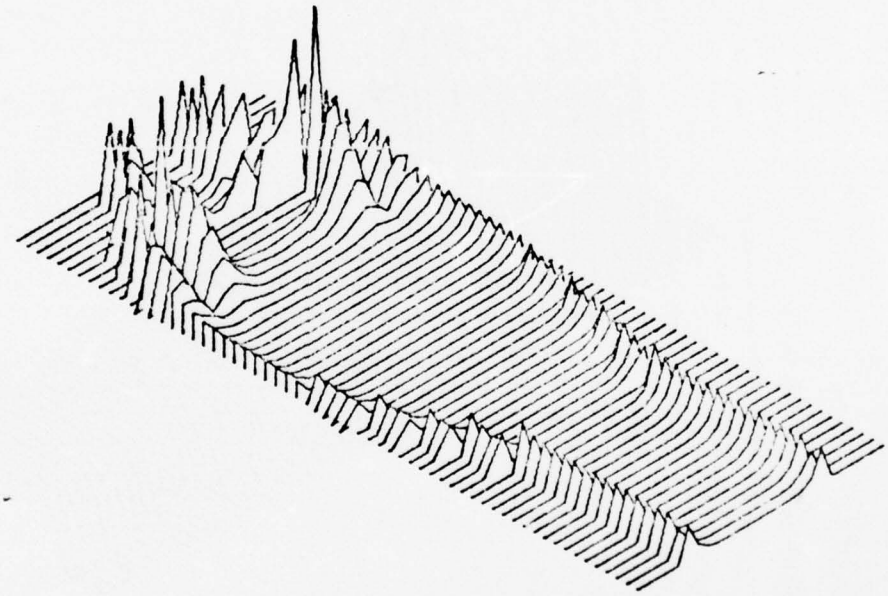
TOTAL VELOCITY

R= 150.

FRAME NO. 15.

Fig. 171: Total Velocity Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position (One Time Frame)





VORTICITY FUNCTION

R= 150.

FRAME NO. 15.

Fig. 172: Vorticity Function Plot for a Disc-Type Artificial Heart Valve at Full Open Position (One Time Frame)

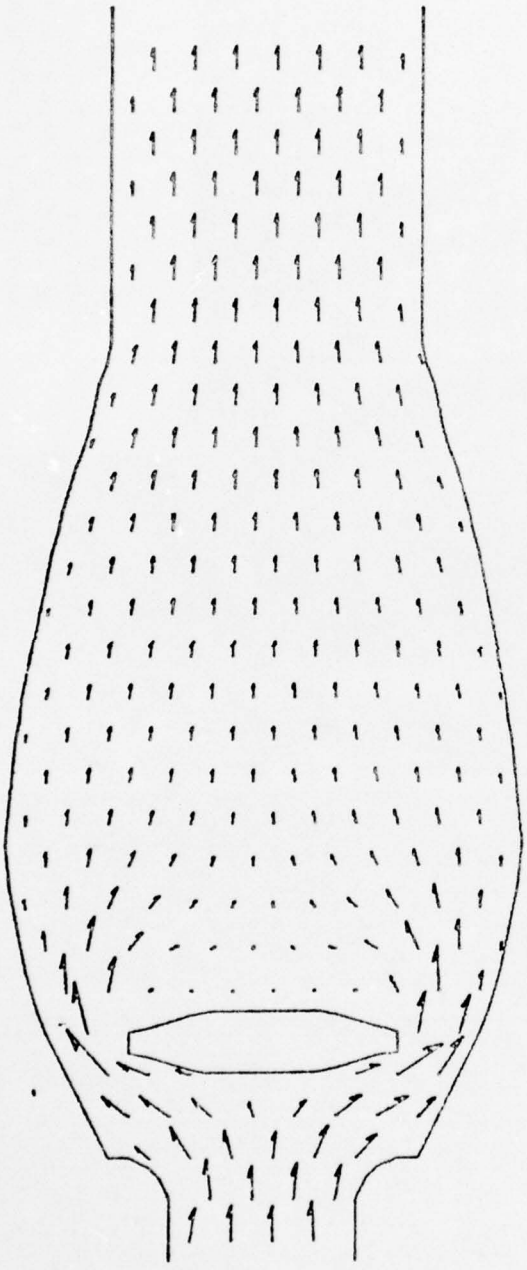


Fig. 173: Velocity Vector Plot for a Disc-Type Artificial Heart Valve at Full Open POsition (One Time Frame)

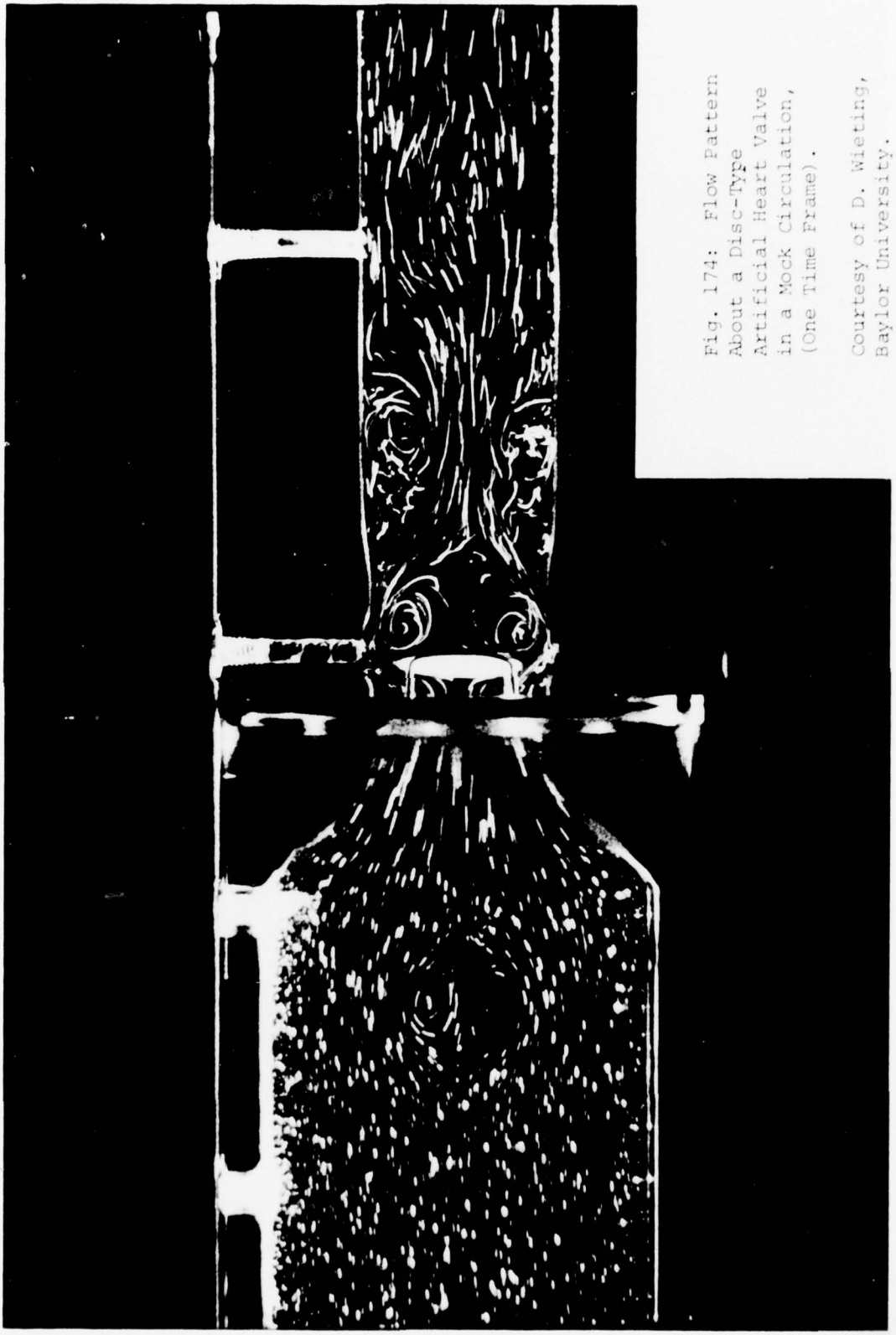


Fig. 174: Flow Pattern About a Disc-Type Artificial Heart Valve in a Mock Circulation, (One Time Frame).

Courtesy of D. Wieting, Baylor University.

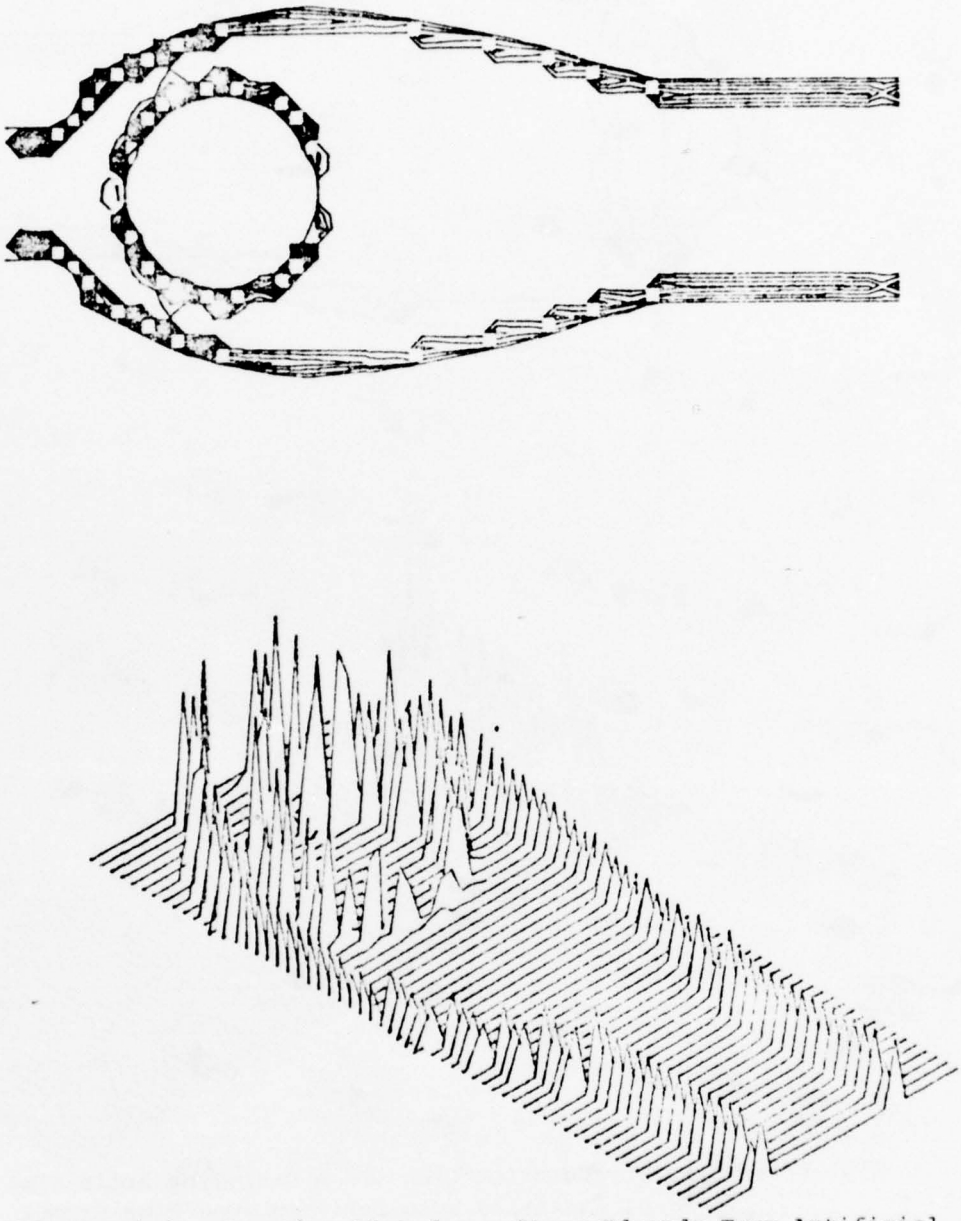


Fig. 175: Vorticity Function Plot for a Starr-Edwards Type Artificial Heart Valve at Partially Open Position (One Time Frame)

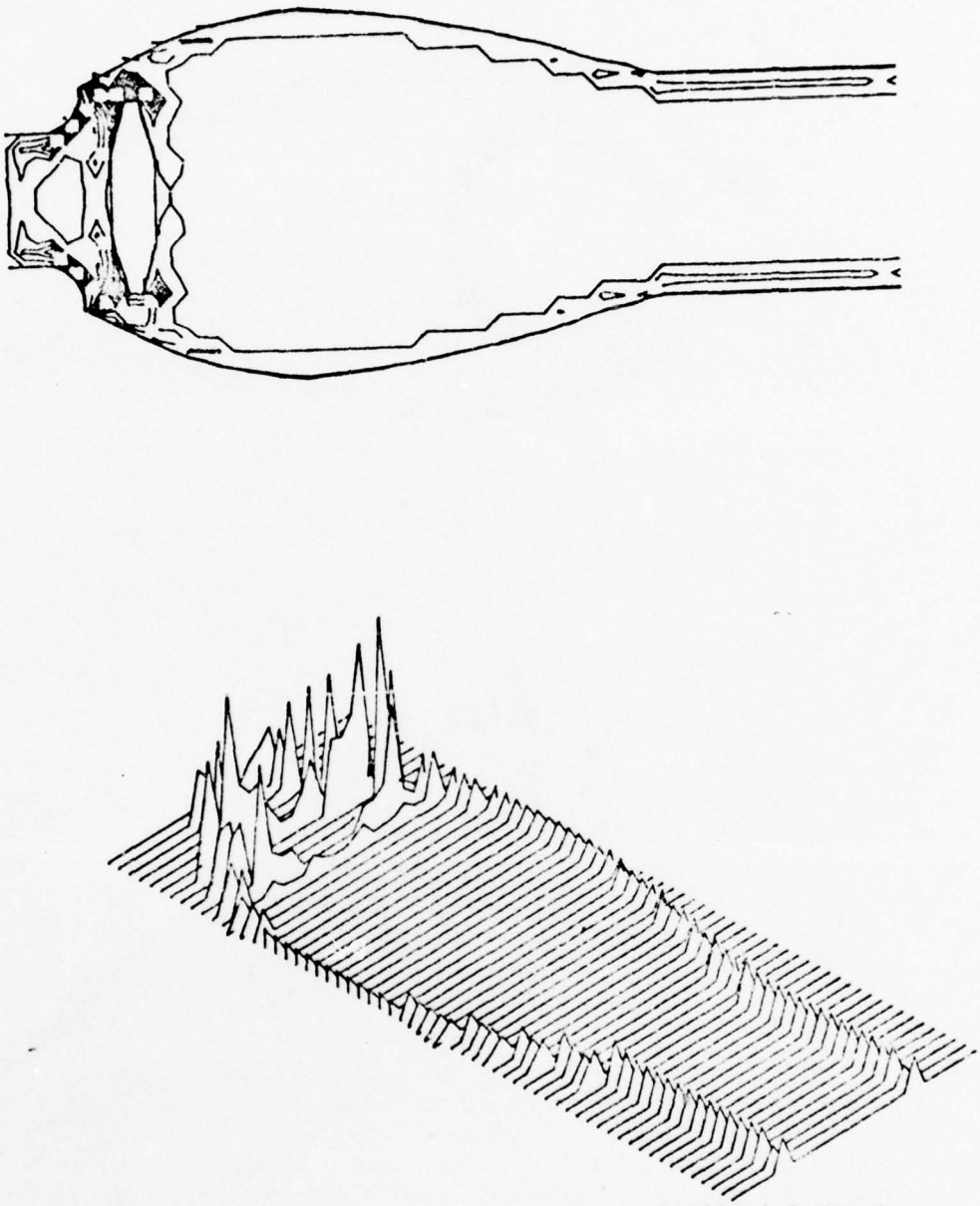


Fig. 176: Vorticity Function Plot for a Disc-Type Artificial Heart Valve at Partially Open Position (One Time Frame)

CHAPTER VII

CONCLUSIONS.

It is obvious that fluid flow experimentation by computer graphics is an extremely versatile tool. Less cost and time are required to study design concepts by computer experiment than by physical experiments. Also, in wind tunnel experiments or in hydraulic modeling, modifications of data interpretation are usually required to compensate for flow velocity uncertainties, viscosity variation and fluid compressibility. Such variations are easily written into or ignored in computer programs. Detailed computer results form working approximations to intractable mathematical concepts.

Although the primary function of this presentation has been accomplished, namely, the viewing of a powerful new approach with subsequent important applications, it is realized that a great deal of effort still remains for true duplication of natural blood flow in vascular channels. This point has been discussed by Greenfield [60, 61, 62]. The fluidal system involves not a steady motion but a pulsatile motion. The channel walls of the arterial tree are not fixed, solid walls, but rather, they display elastomeric or viscoelastic properties. Such properties will be seen in future models formed for graphic display. Other effects, such as tapering of vascular channels and the presence of reflective waves emanating from the pulsatile motion, will serve as future inputs to the overall program. All these effects will serve as the bases for continuing reports in this series.

The computer experiment approach could replace present flow pattern studies formed by dye observations since observations cease when dye and other flow components become mixed, obscuring the flow details. At the

least, the computer experiment would complement the present tool of streaming birefringence. Numerically, in the computer experiment, additional perturbation effects would possibly be injected into the flow stream. These would appear as mathematical analogous forms to serum thrombotic accelerator activity, which appears to enhance thrombus formation in the flowing blood. An attempt could then be made concurrently to change channel boundary conditions to conform to the periodic pulsatile mode with viscoelastic conditions present.

Ancillary to the theoretical studies has been the initiation of experiments for in vitro verification of presently evolving theoretical flow studies. A mock circulation apparatus, bolstered by the addition of photostress equipment, is presently being employed to evaluate fluid flow motion in the transvalvular areas of the heart. With the inclusion of another section, it will also be used to study atheromatosis sites in vascular channels.

There are certainly weaknesses present in the system. Superposition of a curvilinear boundary onto a rectangular finite difference net is tenuous. The truncation error may be enlarged and the presence of the non-symmetrical finite difference stencil can lead to additional error as well as possible propagation of the round-off error. Many flow configurations are complicated, such as the combined renal and abdominal aortic bifurcation sections, which result in many mesh nodes typed as "exterior" to the solution space. Computer storage must then be allocated for these "unused" nodes. Consequently, sufficient detail cannot be allocated to areas of flow where detail is required and such problems as "flexing" boundaries compound the difficulties. Such weaknesses are being attacked by new computer "hardware" studies, in the case of storage problems and "software" studies in all situations.

The storage is presently being tripled in capacity which should allow a finer mesh for greater detailed study at a specific point without losing gross evaluation at point distal to the obstacle. In turn, another view of the problem of a curved boundary has been investigated and it remains to be verified.

Another weakness of the presently discussed technique is that it employs vorticity and stream function values as the primary dependent variables within the computational mesh. The advantages were that only two dependent variables were needed and mass was easily conserved, since the use of a stream function identically satisfied the mass conservation condition. However, the disadvantages associated with the vorticity - stream function method are its poor extension to three dimensions which requires the introduction of a vector potential for velocity, and the difficulty encountered in satisfying free surface boundary conditions. Also, true channel flow remains to be described and pulsating motion and viscoelastic concepts are difficult to investigate with the present nodal parameters. To overcome the difficulties, the Marker and Cell method [63] is being intensely pursued by the second author of this report. This work will be the bases of the second report in this series. The method uses pressure and velocity as primary dependent variables and allows free surfaces; the latter useful for pulsing and viscoelastic effects. It also maintains accuracy with a minimum of computer time through the use of a corrective procedure.

The computer experiment is basically designed to supplement other techniques and to allow new fluid flow investigations. The contribution of the computer graphics approach is the ability to present solutions to certain previously intractable situations and in detail. Hopefully, in the future, solutions to non-steady flow problems will be in the forms of

a computer directed optical display continuum, copied by movie camera, with the allowance of desired changes in flow configuration upon call from computer programming storage banks.

In the future, usage will be made of the head-mounted display, developed by Sutherland [64], and presently being augmented at the University of Utah. Two miniature cathode-ray tubes are mounted on the user's head; one tube on each side of an eye so that prisms then direct and form a stereoscopic view in front of the user's viewpoint. As the user turns his head, the theoretical perspective object appears to remain stationary as the viewer moves into an appropriate viewing position to examine an object's particular feature. It therefore appears feasible for one to project himself into a computer displayed prosthetic heart valve and watch "theoretical" blood rush towards him from the atrium or turn around and view the valve occluder moving away or towards him. Feasible too, would be viewing of flow about an atherosclerotic lesion.

BIBLIOGRAPHY

1. Braunwald, N.S. and D.E. Detmer, "A Critical Analysis of the Status of Prosthetic Valves and Homografts", Progress in Cardiovascular Diseases, Cardiovascular Surgery I, vol. 11, no. 2, Sept., 1968, p. 113.

2. Harken, D.E., "Artificial Heart Valves", Lancet, vol. 2, 1962, p. 763.

3. Hufnagel, C.A. and W.P. Harvey, "Surgical Correction of Aortic Insufficiency: Preliminary Report", Bulletin of Georgetown Univ. Med. Ctr., vol. 6, 1953, p. 60.

4. Starr, A. and M.L. Edwards, "Mitral Replacement: Clinical Experience with a Ball Valve Prosthesis", Annals of Surgery, vol. 154, Oct., 1961, p. 726.

5. Harken, D.E., Soroff, H.S., Taylor, W.J., Lefemine, A.A., Gupta, S.K. and S. Lunzer, "Partial and Complete Prosthesis in Aortic Insufficiency", Journal of Thoracic and Cardiovascular Surgery, vol. 40, 1960, p. 744.

6. Malm, R.J., Bowman, F.O.Jr., Harris, P.D. and A.T.W. Kowalik, "An Evaluation of Aortic Valve Homografts Sterilized by Electron Beam Energy", Journal of Thoracic and Cardiovascular Surgery, vol. 54, 1967, p. 471.

7. Duvoisin, G.E. and D.C. McGoon, "The Advantages and Disadvantages of Prosthetic Valves for Aortic Valve Replacement", Progress in Cardiovascular Diseases, vol. 11, no. 4, Jan., 1969, p. 294.

8. Magovern, G.J., Kent, E.M. and W.B. Cushing, "Sutureless Mitral Valve Replacement", Annals Thoracic Surgery, vol. 2, 1966, p. 474.

9. Davey, T.B., Kaufman, B., Smeloff, E.A. and R.S. Cartwright, "Design of Prosthetic Heart Valves", Publication 65-WA/HUF-5, Amer. Soc. Mech. Engrs., N.Y., 1965.

10. Hufnagel, C.A. and P.W. Conrad, "Comparative Study of Some Prosthetic Valves for Aortic and Mitral Replacement", Surgery, vol. 57, 1965, p. 205.

11. Cross, F.S. and R.D. Jones, "A Caged-lens Prosthesis for Replacement of the Aortic and Mitral Valves", American Thoracic Surgery J.,vol. 2, 1966, p. 499.

12. Kay, J.H., Kawashima, Y., Kagawa, Y., Tsuji, H.K., and J.V. Redington, "Experimental Mitral Valve Replacement With a New Disc Valve", Am. Thoracic Surgery J., vol. 2, 1966, p. 485.

13. Melrose, D.G., Bentall, H.H., MacMillen, I.K.R., Flege, J.B., Alvarez-Diaz, F.R., Nahas, R.A., Fautley, R. and J. Carson, "The Evaluation of a Mitral Valve Prosthesis", Lancet, vol. 2, 1964, p. 623.

14. Gott, V.L., Rowe, G.G., Daggett, R.L., Whiffen, J.D., Koepke, D.E. and W.P. Young, "Preoperative and Postoperative Hemodynamic Data in Patients With a Prosthetic Hinged-Leaflet Aortic Valve", Circulation (Suppl. I), vol. 96, 1965, p. 31.

15. Palmer, T.E., Lautsche, E.V., Sanmarco, M.E. and J.C. Davila, "A Non-Thrombogenic, Non-Anticoagulant-Dependent Mitral Valve Prosthesis", Circulation (Suppl. I), vol. 42, 1967, p. 35.

16. Kaster, R.L., Lillehei, C.W. and P.J.H. Starek, "The Lillehei-Kaster Pivoting Disc Aortic Prosthesis and a Comparative Study of its Pulsatile Flow Characteristics with Four Other Prosthesis", Trans, Amer. Soc. Art. Int. Organs, vol. 16, 1970, p. 233.

17. Wada, J., "A New Hingeless Valve", 2nd Nat. Conf. on Prosthetic Heart Valves, Los Angeles, May 30, 1968.

18. Vorhauer, B.W. and J.W. McElhaney, "The Analysis and Design of a Streamlined Heart-Valve Prosthesis", Presented at Human Factors Conf., Wash. D.C., Mar. 28, 1966, ASME Publication 66-HUF-7.

19. Kaster, R.L., Lillehei, C.W., Helgerud, R.J., Rassman, W.R. and A. Nakib, "The Design Features and the Development of the Toroidal Prosthetic Heart Valve", Trans. Amer. Soc. Artif. Int. Organs, vol. 15, 1969, p. 206.

20. Cross, F.S., Akao, M., and R.D. Jones, "Comparison of Ball and Lens Heart Valve Prostheses", Surgery, Oct., 1967, p. 797.

21. Roberts, W.C., Levinson, G.F. and A.G. Monow, "Lethal Ball Variance in the Starr-Edwards Prosthetic Mitral Valve", Archives of Internal Med., vol. 126, Sept., 1970, p. 517.

22. Texon, M., "The Hemodynamic Basis of Atherosclerosis", Medical Counterpoint, Dec., 1969, p. 11.

23. Wesolowski, S.A., Fries, C.C., Sabini, A.M. and P.N. Sawyer, "Turbulence, Internal Injury, and Atherosclerosis", Biophysical Mechanisms in Vascular Homeostatis and Intravascular Thrombosis, P.N. Sawyer (ed.), Appleton-Century-Crofts, N.Y., 1965, p. 147.

24. Spain, D.M., "Atherosclerosis", Scientific American,

25. Moses, C., Atherosclerosis: Mechanisms as a Guide to Prevention, Lea and Febiger, Philadelphia, 1963, Chapter 2.

26. Carrell, A. and C.C. Guthrie, "Arterio-Sclerose par modification Chirurgicale de la Circulation", Compt. rendue Soc. de Biol., vol. 60, 1906, p. 730.

27. Klotz, O., "Experimentelle Arbeits-Arteriosklerose", Zentralblat. f. allg. Path. and Path. Anat., vol. 19, 1908, p. 535.

28. Anitschkow, N.N., "Uber die Atherosklerose der Aorta beim Kaninchen und uber deren Entstehungsbedingungen", Beitr. z. path. Anat. und z. Allg. Path. vol. 59, 1914, p. 306.

29. Allbutt, C., Diseases of the Arteries, Macmillan, New York, 1915, 2 volumes.

30. Duguid, J.B., "Athoroma of the Aorta", J. Pathology and Bacteriology, vol. 29, 1926, p. 371.

31. Willis, G.C., "Localizing Factors in Atherosclerosis", Canadian Med. Assoc. J., vol. 70, 1954, p. 1.

32. Texon, M., "The Hemodynamic Concept of Atherosclerosis with Particular Reference to Coronary Occlusion", Arch. Int. Med., vol. 99, 1957, p. 418.

33. Texon, M., "The Hemodynamic Concept of Atherosclerosis", Amer. J. Cardiol., vol. 5, 1960, p. 291.

34. Texon, M., A.M. Imparato and J.W. Lord, Jr., "The Hemodynamic Concept of Atherosclerosis; The Experimental Production of Hemodynamic Arterial Disease", Arch. Surg., vol. 80, 1960, p. 47.

35. Texon, M., "Mechanical Factors Involved in Atherosclerosis", Atherosclerotic Vascular Disease, ed. by Brest and Moyer, Appleton-Century-Crofts, New York, 1967, chapter 3.

36. Salvadori, M.G. and M.L. Baron, Numerical Methods in Engineering, Prentice-Hall, Englewood Cliffs, N.J., 1961, chap. 2.

37. Thom, A. and C.J. Apelt, Field Computations in Engineering and Physics, Van Nostrand, London, 1961.

38. Shaw, F.S., Relaxation Methods, Dover, N.Y., 1953.

39. Booy, M.L., "A Noniterative Numerical Solution of Poisson's and Laplace's Equations with Application to Slow Viscous Flow", Paper no. 66-FE-1, ASME-EIC Fluid Engr. Conf., Denver, Colo., 1966.

40. Fromm. J.E., "A Method for Computing Nonsteady Incompressible Viscous Fluid Flows", Los Alamos Sci. Lab. Rpt. LA-2910, 1963.

41. Fromm, J.E. and F.H. Harlow, "Numerical Solution of the Problem of the Vortex Street Development", Physics of Fluids, vol. 6, 1963, p. 975.

42. Dufort, E.C. and S.P. Frankel, "Stability Conditions in the Numerical Treatment of Parabolic Differential Equations", Math. Tables and Other Aids to Computation, vol. 7, 1953, p. 135.

43. Pearson, C.E., "A Computational Method for Viscous Flow Problems", Journal of Fluid Mech., vol. 21, 1965, p. 611.

44. Esch, R.E., "An Alternative Method of Handling Boundary Conditions and Various Computational Experiments in the Numerical Solutions of Viscous Flow Problems", Sperry Rand Res. Rpt. SRRC-RR-64-64, Sudbury, Mass., 1964.

45. Peaceman, D.W. and H.H. Rachford, "The Numerical Solution of Parabolic and Elliptic Equations", J. Soc. Industrial and Appl. Math., vol. 3, 1955, p. 28.

46. Weber, A.H., "An Investigation of Nonsteady Flow Between Parallel Plates", unpublished PhD Thesis, Dept. of Meteorology, Univ. of Utah, 1966.

47. Todd, J. (ed.). Survey of Numerical Analysis, McGraw-Hill, N.Y., 1962.

48. Lawrensen, D.J., "Numerical Methods", Field Analysis, ed. by D. Vitkovitch, Van Nostrand, N.Y., 1966.

49. Schlichting, H., "Laminare Kanaleinlaufstromung", ZAMM, vol, 14, 1934, p. 368.

50. Prandtl, L. and O.G. Tietjens, Applied Hydro-and Aeromechanics, Dover, N.Y., 1934.

51. Schlichting, H., Boundary Layer Theory, McGraw-Hill, N.Y., 1955.

52. Hung, T.K. and E.O. Macagno, "Laminar Eddies in a Two-Dimensional Conduit Expansion", La Houille Blanche, vol. 21, no. 4, 1966, p. 391.

53. Sparrow, G.M., "Experiments on Laminar Flow Development in Rectangular Ducts", paper no. 66-FE-7, ASME-EIC Fluids Engineering Conf., Denver, 1966.

54. Forsythe, G.E. and W.R. Wasow, Finite Difference Methods for Partial Differential Equations, Wiley and Son, New York, 1960.

55. Salvadori, M.G. and M.L. Baron, Numerical Methods in Engineering, Prentice-Hall, Englewood Cliffs, N.J., 1961.

56. Todd, J., Survey of Numerical Analysis, McGraw-Hill, N.Y., 1962.

57. Greenfield, H. and C. Brauer, "Hemodynamic Studies Involving A Computer Simulation Technique", Presented at International Symposium of Computer Aided Design, Southampton, England, Apr., 1969 (published as Inst. of Electronic Engrg. Publication no. 51).

58. Greenfield, H., C. Brauer, K. Reemtsma and W. Kolff, "A New Approach for Analyses of Vascular Degenerative Processes", Jane Nugent Cochems Medical Competition Paper (internal report), Jan., 1969.
College of Medicine
Univ. of Utah

59. Wieting, D.W., "Dynamic Flow Characteristics of Heart Valves", unpublished Doctoral Thesis, University of Texas, May, 1969.

60. Greenfield, H. and K. Reemtsma, "Preliminary Analysis of Blood Flow Characteristics in the Abdominal Aorta by Computer Interpretation", Presented at 1970 Brunel International Graphics Symposium, Uxbridge, England, April, 1970.

61. Reemtsma, K., Sandberg, L. and H. Greenfield, "Some Theoretical Aspects of Vascular Degeneration", American Journal of Surgery, vol. 119, May, 1970, p. 548.

62. Greenfield, H., "Rheologic Factors in Vascular Degeneration: The Use of Mathematical Theory and Computer Graphics", Journal of Computers and Biomedical Research, pending.

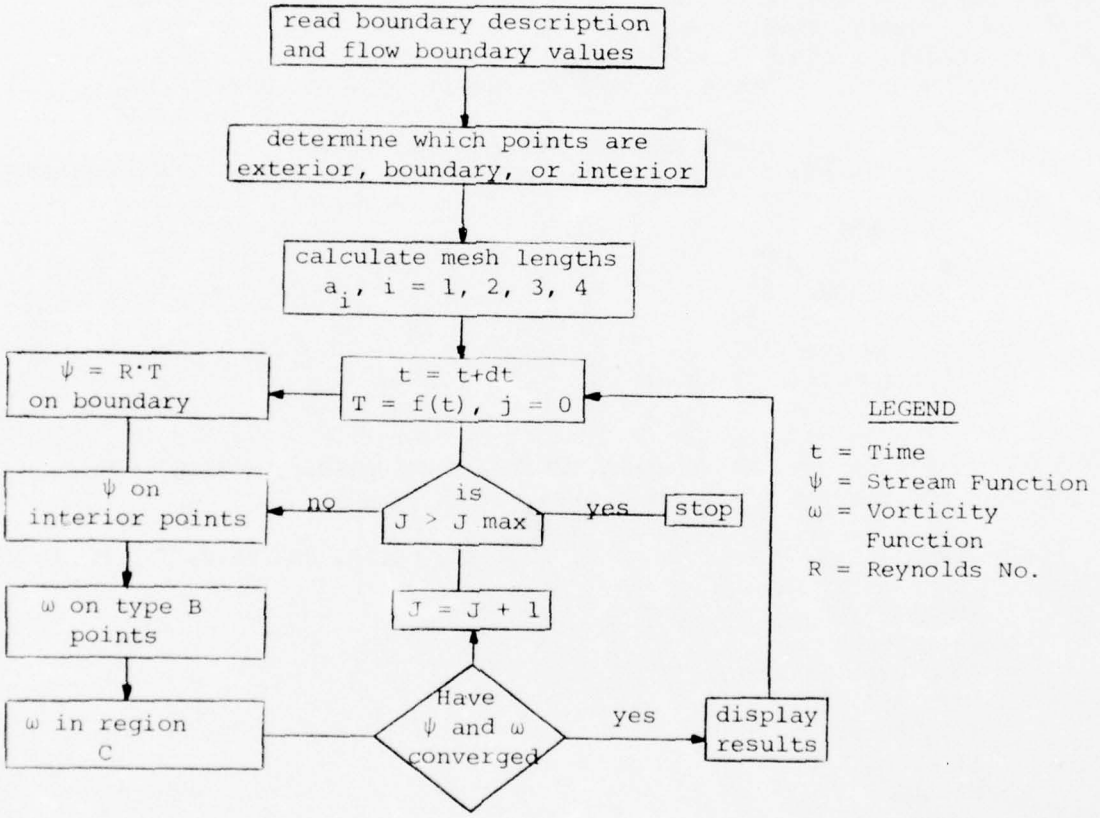
63. Welch, J.E., Harlow, F.H., Shannon, J.P. and B.J. Daly, "The MAC Method: A Computing Technique for Solving Incompressible, Transient, Fluid Flow Problems Involving Free Surfaces", Los Alamos Sci. Lab. Rpt., LA-4325, 1966.

64. Sutherland, I., "A Head-Mounted Three Dimensional Display", Proc. of Fall Joint Computer, Cmt., 1968, p. 757.

PRECEDING PAGE BLANK-NOT FILMED

APPENDIX:

FLOW CHART AND COMPUTER PROGRAM



```
@I FOR XYMAIN,XYMAIN
      DIMENSION SN(61,31), SO(61,31), VN(61,31), VO(61,31),
     1  TA(61,31), IFP(2),
     2 BND(61,31), A(61,31), TB(61,31)
      DIMENSION S1(61,31), S2(61,31), S3(61,31), S4(61,31)
      DIMENSION BPF(100,5), BPS(100,5), IBF(100,5), JBF(100,5),
     1          IBS(100,5), JBS(100,5), NBPF(6), NBPS(6)
      DIMENSION IVS(250), IVB(250), JVB(250)
      DIMENSION ISB(250), JSB(250), JSE(250)
      DIMENSION ISB1(250), JSB1(250), ISE1(250)
      DIMENSION XBP(100,5), YBP(100,5), BLN(5)
      DIMENSION EV(60), ES(60), ITERS(60)
      DIMENSION BP1(5)
      COMMON/COM3/ ISB, JSB, JSE, KSB, ISB1, ISE1, JSB1, KSB1
      COMMON/COM1/ BND, M, N, M1, N1, M2, N2, M3, N3
      COMMON/COM2/ A
      INTEGER BND, BLANK, SWITCH, VTOBIG, DISPLY, EDIS, PN, PNCH,
     1   STEADY, SSTATE, REITER
      LOGICAL RESTRT, REDO, EPLOT,
      COMMON/INTRC/ OMEGA, R, NFRAME, SFVBND, SFVORT, BPR1, BPR2,
     1              BPR3, BPR4, DISPLY, DR, DT, EVOR, ESTR, IWRITE,
     2              NIPR, RESTRT, REDO, EPLOT, SWAPON
      COMMON RELAY/ IFP, NFILE, XBP,YBP,BLN,BPF,BPS,IBF,JBF,IBS,JBS,NBPF,NBPS

      DATA BLANK/6H      /
      DATA PN/'PN'/
      DATA STEADY/'SS'/

      TTIME = EXTIME(1)
         CALL DATA(M,N,END,A)
     1

      READ 1, R, DR, DT, NFRAME, IWRITE, IPRS, DISPLY, SSTATE, PNCH,
     1        BPR1, BPR2, BPR3, BPR4, OMEGA, DOMEGA,
     2        SFVBND, SFVBND1, SFVORT
    1 FORMAT(3F10.5, 4I5, 4X, 4A2, I2/4(5X, F5.1), 6X, F5.2, 7X, F5.3/
     1      3(7X, F5.2))
      IF(DISPLY.EQ.0) GO TO 2
      CALL RELOAD
      CALL TYPLIN('READY␣Δ')
    2 NIPR = 1
      CALL NOSKIP
      EPLOT = .TRUE.
      PSCALE = 1.0
      JJMAX = 60
      MSP = 1
      NSP = 1
      IF(N.GT.31) NSP = 2
      N1 = N-1
      N2 = N-2
      N3 = N-3
      M1 = M-1
      M2 = M - 2
      M3 = M - 3
      H = 1./FLOAT(N1)
```

```
      H2 = H*H
      DH = FLOAT(N1)
      EVOR = .0750*DR
      ESTR = 0.000050*DR
      DTP = 0.0001
      DT1 = 2.*H2/DT
      DTDX = DT/H2
      IPSKIP = 0

      DO 3 I=2,M1
      DO 3 J-2,N1
      S1(I,J) = 1./(A(I+1,J)*(A(I+1,J) + A(I-1,J)))
      S2(I,J) = 1./(A(I-1,J)*(A(I+1,J) + A(I-1,J)))
      S3(I,J) = 1./(A(I,J+1)*(A(I,J+1) + A(I,J-1)))
    3 S4(I,J) = 1./(A(I,J-1)*(A(I,J+1) + A(I,J-1)))

    4 DO 5 J=1,N
      DO 5 I=1,M
      SN(I,J) = 0.0
      SO(I,J) = 0.0
      VN(I,J) = 0.0
      TA(I,J) = 0.0
      TB(I,J) = 0.0
    5 VO(I,J) = 0.0
      RESTRT = .FALSE.
      REDO = .FALSE.
      BP1(1) = BPR1
      BP1(2) = BPR2
      BP1(3) = BPR3
      BP1(4) = BPR4
      BP1(5) = BPR5

C        STEADY STATE DATA....

      IF(SSTATE.NE.STEADY) GO TO 9
      READ 6, ((SN(I,J), VN(I,J), I=1M), J=1,N)
    6 FORMAT(9F8.1)
    7 DO 8 I=1,M
      DO 8 J=1,N
      SO(I,J) = SN(I,J)
      TA(I,J) = SN(I,J)
      VO(I,J) = VN(I,J)
    8 TB(I,J) = VN(I,J)

C     UNIT 14 IS A FAST DRUM...USED FOR TEMPORARY STORAGE

    9 REWIND 14
      WRITE(14) TA, TB,

      CALL VBND1(IVS,IVB,JVB,KVB)
```

```
      PRINT 11, D, M, N, DT, TMAX, SEVORT, SEVBND, EVOR, ESTR,
     1   JJMAX, H, DTP, DTDX, SEVBD1, OMEGA, DOMEGA
   11 FORMAT(1H1,//20X, 18HPROGRAM PARAMETERS//20X, 2HN=I3//20X, 2HM=I3
     1//20X, 2HR=F10.1//20X, 3HDT=F9.5//20X, 5HTMAX=F9.5//20X,
     2 'SEVORT=' F14.7//20X, 'SEVBND=' F14.7//20X, 5HEVOR= F14.7//20X,
     3  5HESTR=F14.7//20X, 6HJJMAX= I4//20X,  2HH= F11.7// 20X,
     4 4HDTP=, F14.7// 20X, 'DT/H2=', F14.7// 20X, 'SEBD1=', F14.7,
     5  //20X, 'OMEGA=', F14.7, //20X, 'DOMEGA=', F14.7, 1H1)

      IF(IGUESS.NE.INGUES) GO TO 16

      REWIND 4
      READ(4) M, N, NFILE, DT, TMAX, R, OLDDR
      NWORD = N+1
      CALL INOUT(1,4,BND,NWORD)
      CALL INOUT(1,4,A,NWORD)
      READ(4) IBF, JBF, NDPF
      DO 12 I=1,NFILE
      READ(4) IK, (XBP(K,I), YBP(K,I), K=1,IK)
   12 BLN(I) = IK
      SWITCH = 1
      IPSW = 1
      NTIME = 0
      R1 = 0.0
      T = 0.0
      TF = 0.0
      DO 13 K=1,NSKIP,3
      IPSKIP = IPSKIP + 3
      IF(IPSKIP.GE.IPRS) IPSKIP = 0
      NTIME = NTIME + 3
      T = T + 3.0*DT
      TF = TF + 3.0*OLDDR/R
      R1 = R1 + 3.0*OLDDR
      CALL INOUT(1,4,TA,NWORD)
      CALL INOUT(1,4,TB,NWORD)
      REWIND 14
      WRITE(14) TA, TB, TC, TD
      CALL INOUT(1,4,SO,NWORD)
      CALL INOUT(1,4,VO,NWORD)
      CALL INOUT(1,4,SN,NWORD)
      CALL INOUT(1,4,VN,NWORD)
   13 CONTINUE
      IF(PERT.NE.PI) R = R1
      CALL OUTFLO
      PRINT 14
   14 FORMAT(1H0, 20X, '.... INITIAL GUESS .....',///)
      CALL PRINT(H,DT,TMAX,R,MSP,NSP,SN,VN,Z,ITER,JJ,T,IP,BND,M,N,
     1 M1,N1,1,IBF,JBF,NDPF)
      IF(SSTATE.NE.STEADY) GO TO 20
      DO 15 I=1,M
      DO 15 J=1,N
      SO(I,J) = SN(I,J)
      TA(I,J) = SN(I,J)
      VO(I,J) = VN(I,J)
   15 TB(I,J) = VN(I,J)
      REWIND 14
      WRITE(14) TA, TB, TC, TD
      GO TO 19
```

```
      GO TO (17,19), IWRITE
   17 REWIND 4
      WRITE (4) M, N, NFILE, DT, TMAX, R, DR
      NWORD = N*M
      CALL INOUT(0,4,END,NWORD)
      CALL INOUT(0,4,A,NWORD)
      WRITE(4) IBF, JBF, DBF
      DO 18 I=1,NFILE
      IK = ILN(I)
   18 WRITE (4) IK, (XBP(K,I), YBP(K,I), K=1,IK)

   19 SWITCH = 1
      IPSW = 1
      T = 0.0
      TF = 0.0
      NTIME = 0
      IF(SSTATE.NE.STEADY) GO TO 20
      NTIME = 1
      TF = 1.
      T = R*DTP/DR
      DT1 = 2.*H2/DTP
      DT = DTP
      DR = 0.0
      GO TO 27

C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

   20 JJ = 0
      REITER = 0
      IF(STEADY.EQ.SSTATE) REITER = 1
      TTCOMP = 0.0
      TTVBNE = 0.0
      TTVORT = 0.0
      TTSTR = 0.0
      TTMISC = 0.0
      T = T + DT
      TF = TF + DR/R
      NTIME = NTIME + 1
      ESUMT = 0.5
      IF(NTIME.LE.NFRAME) GO TO 27

      GO TO (21,24), SWITCH
   21 DT1 = 2.*H2/DTP
      DT = DTP
      T = T + 2.*DTP
      NFRAME = 20
      DR = 0.0
      TF = 1.
      SWITCH = 2
      IF(PN.NE.PNCH) GO TO 23
   22 CALL CRDIDS
      PUNCH 6, ((SN(I,J), J=1,J), I=1,M), J=1,N)
      CALL CRDIDF
      CALL EXIT
```

```
   24 GO TO (25,26), IWRITE
   25 CALL NTRAN(4,9,11)
   26 CALL EXIT

   27 DO 29 KK=1,NFILE
      K2 = NBPS(KK)
      IF(K2,EQ.0) GO TO 29
      DO 28 K=1,K2
      I = IBS(K,KK)
      J = JBS(K,KK)
   28 SN(I,J) = BPS(K,KK)*TF*BP1(KK)*R
   29 CONTINUE
      IF(NTIME.EQ.1) CALL OUTFLO
      DO 31 KK=1,NFILE
      K1 = NBPF(KK)
      IF(K1.EQ.0), GO TO 31
      DO 30 K=1,K1
      I = IBF(K,KK)
      J = JBF(K,KK)
   30 SN(I,J) = BPF(K,KK)*TF*R
   31 CONTINUE
      IF(REITER.GT.0) GO TO 37

C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C          COMPARISON OF TIME ADVANCE BY ADAMS-BASHFORTH 4-POINT FORMULA
C          AND QUADRATIC EXTRAPOLATION ......
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

   32 REWIND 14
      READ(14) TA, TB

      DO 33 I=2,M1
      DO 33 J=2,N1
      IF(BND(I,J).GE.1) GO TO 33
      TC = TA(I,J)
      TD = TB(I,J)
      TA(I,J) = SO(I,J)
      TB(I,J) = VO(I,J)
      SO(I,J) = SN(I,J)
      VO(I,J) = VN(I,J)
      GO TO (34,33), IPSW
   34 SN(I,J) = 3.0*SO(I,J) - 3.0*TA(I,J) + TC
      VN(I,J) = 3.0*VO(I,J) - 3.0*TB(I,J) + TD
   33 CONTINUE

   37 REWIND 14
      WRITE(14) TA, TB
```

```
C         IN THIS TEST... ADVANCE BY ADAMS-BASHFORTH ONLY     217
C         THE TYPE 'C' POINTS.........

      IF(BND(I,J-1).EQ.1 .OR. BND(I,J+1).EQ.1 .OR.
     1   BND(I+1,J).EQ.1 .OR. BND(I-1,J).EQ.1) GO TO 35


      FN0 = (VO(I,J+1) + VO(I,J-1) + VO(I+1,J) + VO(I-1,J) - 4.*VO(I,J)
     1         - 0.25*(VO(I+1,J) - VO(I-1,J))*(SO(I,J+1) - SO(I,J-1))
     2         + 0.25*(VO(I,J+1) - VO(I,J-1))*(SO(I+1,J) - SO(I-1,J)))/H2

      FN1 = (TB(I,J+1) + TB(I,J-1) + TB(I+1,J) + TB(I-1,J) - 4.*TB(I,J)
     1         - 0.25*(TB(I+1,J) - TB(I-1,J))*(TA(I,J+1) - TA(I,J-1))
     2         + 0.25*(TB(I,J+1) - TB(I,J-1))*(TA(I+1,J) - TA(I-1,J)))/H2

      FN2 = (TD(I,J+1) + TD(I,J-1) + TD(I+1,J) + TD(I-1,J) - 4.*TD(I,J)
     1         - 0.25*(TD(I+1,J) - TD(I-1,J))*(TC(I,J+1) - TC(I,J-1))
     2         + 0.25*(TD(I,J+1) - TD(I,J-1))*(TC(I+1,J) - TC(I-1,J)))/H2

      VN(I,J) = VO(I,J) + (23.0*FN0 - 16.0*FN1 + 5.0*FN2)*DT/12.0
      GO TO 36

   35 VN(I,J) = 3.0*VO(I,J) - 3.0*TB(I,J) + TD(I,J)
   36 CONTINUE

      REWIND 13
      WRITE(13) VN

   38 RR = R*TF*PSCALE
      IF(NFILE.GE.3) CALL OUTFILE
      IF(.NOT. SWAPON) PRINT 39, NTIME, RR
   39 FORMAT(1H0//30X, 22HERROR TERMS FOR NTIME=, I5, 3X, 2HR=F6.0/)
      IF(.NOT. SWAPON) PRINT 40
   40 FORMAT(1H0,  3X, 5H ITER, 10X, 10HVORTICITY , 15X, 6HSTREAM,
     1  4X, 8HSOR ITER, 4X, 9HSOR PARAM, 7X, 10H ERROR SUM, 3X,
     2  'TSTR', 4X, 'TVORT', 4X, 'TVEND'//)
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

C         .... MAIN LOOP ....

      IR = R*TF + .5
      JJ = 0
      EDIS = 1
      NFL = 0
   41 IF(DISPLY.EQ.1) CALL INTRUP
      IF(RESTRT) GO TO 4
      IF(REDO) CALL AGAIN
      DO 42 I=1,M
      DO 42 J=1,N
      TB(I,J) = SN(I,J)
   42 TA(I,J) = VN(I,J)

      JJ = JJ + 1
      TMISC = EXTIME(1)
      CALL STREAM(SN,VN,H2,ITER,ES,OMEGA,ENEW,JJ,ESUMT,A,S1,S2,S3,S4,
     1            DOMEGA,OMEGA1,ESTR)
      TSTR = EXTIME(1)
      CALL VBND(SN,VN,IVB,JJ,IVS,KVL,SFVBND,H2,A,SFVBD1,S1,S2,S3,S4)
```

```
      TVBND = EXTIME(1)
      CALL VORT(VN,VO,SN,SO,T1,H,EH,SFVORT,MIPR)
      TVORT = EXTIME(1)
      TTMISC = TTMISC + TMISC
      TTVBND = TTVBND + TVBND
      TTVORT = TTVORT + TVORT
      TTSTR = TTSTR + TSTR
      TTIME =(TMISC + TVBND + TVORT + TSTR)/60. + TTIME
      TTCOMP = TTVBND + TTVORT + TTSTR

      EES = 0.0
      EEV = 0.0
      DO 43 I=1,M
      DO 43 J=1,N
      EEV = AMAX1(EEV, ABS(TA(I,J)-VN(I,J)))
   43 EES = AMAX1(EES, ABS(TB(I,J)-SN(I,J)))
      DO 45 I=1,M
      DO 45 J=1,N
      IF(BND(I,J).GT.1) GO TO 44
      TA(I,J) = ABS(TA(I,J)-VN(I,J))
      GO TO 45
   44 TA(I,J) = 0.0
   45 CONTINUE
      IF(EEV.LT.EVOR .AND. EES.LT.ESTR) GO TO 49
      VTOBIG = 1
      IF(EEV.GT.10.E10) VTOBIG = 2

      IF(.NOT. SWAPON)
     1  PRINT 46, JJ, EEV, EES, ITER, OMEGA1, ENEW, TSTR, TVORT, TVBND
   46   FORMAT(1H , 5X, I5, F20.6, 3X, F20.6, 3X, I5, 5X, F10.6, 5X,
     1    F12.6, 3F8.2)

      EV(JJ) = EEV
      ITERS(JJ) = ITER
      ES(JJ) = EES
      IF(EDIS.EQ.0) GO TO 47
      IF(DISPLY.EQ.1 .AND. EPLOT) CALL ERRDIS(NTIME,JJ,ITERS,EV,ES,IR,
     1 EVOR,ESTR,M,N,BLN,XBP,YBP,NFILE,TTIME,TVBND,TVORT,TSTR,TMISC,TA,
     2  TTCOMP,TTMISC,TTVORT,TTVBND,TTSTR,NFL)

   47 GO TO (48,49), VTOBIG
   48 IF(JJ.LT.JJMAX) GO TO 41
      CALL EXIT
   49 IF(SWAPON) GO TO 54
      PRINT 46, JJ, EEV, EES, ITER, OMEGA1, ENEW, TSTR, TVORT, TVBND
      PRINT 50, TTSTR, TTVBND, TTVORT, TTCOMP
   50 FORMAT(1H0, 20X, 'TOTAL TIME (SEC)'/ 24X, 'STREAM=', F7.2/10X,
     1  'VORTICITY (BOUNDARY)=', F7.2/10X, 'VORTICITY (INTERIOR)=' F7.2,
     2  /25X, 'TOTAL=', F7.2)

      REWIND 13
      READ(13) TA
      TEM = 0.0
      DO 51 I=2,M1
      DO 51 J=2,N1
      IF(BND(I,J).NE.0) GO TO 51
      IF(TEM.GT.ABS(VN(I,J)-TA(I,J))) GO TO 51
      TEM = ABS(VN(I,J) - TA(I,J))
      IV = I
```

```
      JV = J
   51 CONTINUE
      PRINT 52, TA(IV,JV), VN(IV,JV), TEM, IV, JV
   52 FORMAT(1H0,  9X, 'PREDICTED (A.S.)=', F10.3/20X,
     1  'ACTUAL=', F10.3/1X, 'MAX DIFF=', F10.3, ' AT POINT ',
     2  I2, ',', I2)

      IF(VTOBIG.GT.1) CALL EXIT

C      PRINT AND STORE DATA.....

      IPRINT = 0
      IF(NTIME.LE.1 .AND. REITER.EQ.0) GO TO 56
      IPSKIP = IPSKIP + 1
      IF(IPSKIP.LT.IPRS) GO TO 55
      IPRINT = 1
      IPSKIP = 0
   55 CALL PRINT(M,DT,TMAX,NS,NSP,NSP,SN,VN,IWRITE,ITER,JJ,T,IP,BND,M,N,
     1            M1,N1,IPRINT,IF,IBF,JBF,NBFF,TD)


C      CALCULATE EXIT FLOW PROFILE...

      CALL OUTFLO

C        WRITE VALUES OF STREAM AND VORTICITY FUNCTION ON DRUM
C        TO SAVE IN CASE RE-INITIALIZATION IS DESIRED....

      REWIND 15
      WRITE(15) SN, VN
      REDO = .FALSE.
      REITER = REITER + 1
      IF(NTIME.LE.1 .AND. REITER.LE.1) GO TO 27
      GO TO 28


C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      SUBROUTINE AGAIN

C        THIS ROUTINE RE-INITIALIZES THE STREAM AND VORTICITY
C        FUNCTIONS TO THEIR VALUES AT THE BEGINNING OF THE TIME STEP

      REWIND 15
      READ(15) SN, VN
      JJ = 0
      REDO = .FALSE.
      RETURN


      SUBROUTINE PERTRB
      PERT = 'NP'
      IPERT = IPERT + 1
      IF(IPERT.GT.1) RETURN
      VNMAX = 0.0
      DO 1 I=1,M
      DO 1 J=1,N
      IF(ENE(I,J).NE.0) GO TO 1
      VNMAX = AMAX1(VNMAX,VN(I,J))
    1 CONTINUE
```

```
      VN(16,I1) = 1.5*VNMAX
      VN(17,I1) = 1.5*VNMAX
      RETURN

      SUBROUTINE OUTFLO
C        ....LOWER....
      IF(NFILE.LT.3) GO TO 4
      I = NBPF(3)
      IF(I.LE.0) GO TO 4
      I1 = IBF(1,3)
      I2 = IBF(I,3)
      IF(NTIME.GT.1 .OR. JJ.GT.1) GO TO 2
      TEM1 = SN(I1,2)
      TEM2 = SN(I1,3)
      TEM = (SN(I2,2) - SN(I1,2))/FLOAT(I2-I1)
      DO 1 I=I1,I2
      SN(I,3) = TEM2 + TEM*FLOAT(I-I1)
    1 SN(I,2) = TEM1 + TEM*FLOAT(I-I1)
    2 K = 0
      DO 3 I=I1,I2
      K = K + 1
      SN(I,1) = SN(I,2)
      BPF(K,3) = SN(I,1)/(TF*R)
    3 VN(I,1) = VN(I,2)
C        ....UPPER....
    4 IF(NFILE.LT.4) GO TO 8
      I = NBPF(4)
      IF(I.LE.0) GO TO 8
      I1 = IBF(1,4)
      I2 = IBF(I,4)
      IF(NTIME.GT.1 .OR. JJ.GT.1) GO TO 6
      TEM1 = SN(I1,N1)
      TEM2 = SN(I1,N2)
      TEM = (SN(I2,N1) - SN(I1,N1))/FLOAT(I2-I1)
      DO 5 I=I1,I2
      SN(I,N2) = TEM2 + TEM*FLOAT(I-I1)
    5 SN(I,N1) = TEM1 + TEM*FLOAT(I-I1)
    6 K = 0
      DO 7 I=I1,I2
      K = K + 1
      SN(I,N) = SN(I,N1)
      BPF(K,4) = SN(I,N)/(TF*R)
    7 VN(I,N) = VN(I,N1)
C        ....RIGHT....
    8 I = NBPF(2)
      J1 = JBF(1,2)
      J2 = JBF(I,2)
      IF(NTIME.GT.1 .OR. JJ.GT.1) GO TO 10
      TEM1 = SN(M,J1)
      TEM = (SN(M,J2) - SN(M,J1))/FLOAT(J2-J1)
      DO 9 J=J1,J2
      SN(M1,J) = TEM1 + TEM*FLOAT(J-J1)
    9 SN(M2,J) = TEM1 + TEM*FLOAT(J-J1)
   10 MB = M - 2
      DO 11 J=J1,J2
      DO 11 I=MB,M1
      SN(I+1,J) = SN(I,J)
   11 VN(I+1,J) = VN(I,J)
      K = 0
```

```
      DO 12 J=J1,J2
      K = K + 1
   12 BPF(K,2) = SN(M,J)/(TF*R)

C        ... LEFT ...
      IF(PERTB) CALL PERTRB
      RETURN

C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

      SUBROUTINE OBSTLE
      K1 = 2
    1 K1 = K1 + 1
      IF(K1.GT.NFILE) RETURN

C        CHECK FOR INTERIOR OBSTACLES.........

      L = NLN(K1)
      IF(IFIX(XBP(1,K1)).EQ.1 .OR. IFIX(XBP(L,K1)).EQ.M .OR.
     1   IFIX(YBP(1,K1)).EQ.1 .OR. IFIX(YBP(L,K1)).EQ.N) GO TO 1
      K = K1
      IREF = 1000
      JJ = NLN(K)
      DO 2 J=1,JJ
    2 IREF = MIN0(IREF, IFIX(XBP(J,K)))
      TEM = 0.0
      JREF = 0
      JJ1 = JJ - 1
      DO 3 J=1,JJ1
      IF(IFIX(XBP(J,K)).NE.IREF) GO TO 3
      TEM = TEM + 1.
      JREF = JREF + IFIX(YBP(J,K))
    3 CONTINUE
      TEM1 = FLOAT(JREF)/TEM
      JREF1 = TEM1
      IF(TEM1-FLOAT(JREF1).GT.0) GO TO 4
      JREF2 = JREF1
      GO TO 5
    4 JREF2 = JREF1 + 1
    5 IF(NTIME.GT.100) GO TO 6
      SNREF1 = SN(1,JREF1)
      SNREF2 = SN(1,JREF2)
      GO TO 8
    6 DIREF = 1.0
      DO 7 J=1,JJ
    7 DIREF = AMIN1(DIREF, ABS(XBP(J,K)-FLOAT(IREF)))
      IF(DIREF.LT.0.0001) IREF=IREF-1
      B = A(IREF+1,JREF1)
      SNREF1 = .5*B*(1.+B)*SN(IREF-2,JREF1) - B*(2.+B)*SN(IREF-1,JREF1)
     1 + .5*(1.+B)*(2.+B)*SN(IREF,JREF1)
      B = A(IREF+1,JREF2)
      SNREF2 = .5*B*(1.+B)*SN(IREF-2,JREF2) - B*(2.+B)*SN(IREF-1,JREF2)
     1 + .5*(1.+B)*(2.+B)*SN(IREF,JREF2)
    8 I1 = 1000
      I2 = 0
      J1 = 1000
      J2 = 0
      DO 9 J=1,JJ
      I1 = MIN0(I1, IFIX(XBP(J,K)))
```

```
      I2 = MAX0(I2, IFIX(XBP(J,K)))
      J1 = MIN0(J1, IFIX(YBP(J,K)))
    9 J2 = MAX0(J2, IFIX(YBP(J,K)))
      I1 = I1 - 1
      I2 = I2 + 1
      J1 = J1 - 1
      J2 = J2 + 1
      DO 10 I=I1,I2
      DO 10 J=J1,J2
      IF(BND(I,J).NE.1) GO TO 10
      SN(I,J) = .5*(SNREF1 + SNREF2)
   10 CONTINUE
      GO TO 1

      END
```

BEST AVAILABLE COPY

```
VI FOR PRINT,PRINT
      SUBROUTINE PRINT(DX,DT,TMAX,R,MSP,NSP,SN,VN,IWRITE,IS,JJ,T,IP,
     1                 BND,M,N,M1,N1,IPRINT,TF,IBF,JBF,NBPF,TA)                223
      DIMENSION IBF(100,4), JBF(100,4), NBPF(6), TA(61,26)
      DIMENSION SN(61,26), VN(61,26), BND(61,26), KOUT(61)
      INTEGER BND, BLANK
      DATA BLANK/6H      /

      NWORD = M*N

      GO TO (1,6), IPRINT
    1 P = SCALE(SN,M,N)
      PRINT 2, IS, JJ, P, T, R
    2 FORMAT(1H0, 37X, 22HSTREAM FUNCTION VALUES/37X,15HAV. STREAM ITER.
     1 I5, 13H  VORT. ITER=I5, 3H P=F9.2, 3H T=F9.5/37X, 2HR=F8.0)
      DO 4 I=1,M,MSP
      DO 3 J=1,N,NSP
      K1 = P*SN(I,J) + .5
      KOUT(J) = INBCD(K1)
    3 IF(BND(I,J).EQ.2) KOUT(J) = BLANK
    4 PRINT 5, (KOUT(J), J=1,N,NSP)
    5 FORMAT( 1H , 5X, 31A4)
    6 GO TO (7,8), IWRITE
    7 WRITE(4) P
      CALL INOUT(0,4,SN,NWORD)

    8 GO TO (9,13), IPRINT
    9 VN(1,1) = VN(2,1)
      VN(1,N) = VN(2,N)
      P = SCALE(VN,M,N)
      PRINT 10, P
   10 FORMAT(1H0, 37X, 18HVORTICITY FUNCTION, 3X, 2HP=F9.2//)
      DO 12 I=1,M,MSP
      DO 11 J=1,N,NSP
      K1 = P*VN(I,J) + .5
      KOUT(J) = INBCD(K1)
   11 IF(BND(I,J).EQ.2) KOUT(J) = BLANK
   12 PRINT 5, (KOUT(J), J=1,N,NSP)
   13 GO TO (14,15), IWRITE
   14 CALL INOUT(0,4,VN,NWORD)

   15 GO TO (16,32), IPRINT
   16 DO 18 I=1,M
      DO 18 J=1,N
      IF(BND(I,J).EQ.0) GO TO 17
      TA(I,J) = 0.0
      GO TO 18
   17 TA(I,J) =  .5/DX*(SN(I,J+1) - SN(I,J-1))
   18 CONTINUE
      CALL BNDARY(M,N,M1,N1,IBF,JBF,NBPF,TA)

      P = SCALE(TA,M,N)
      PRINT 19, P
   19 FORMAT(1H0, 37X, 19HU VELOCITY FUNCTION, 3X, 2HP=F9.2//)
      DO 21 I=1,M,MSP
      DO 20 J=1,N,NSP
      K1 = P*TA(I,J)
      KOUT(J) = INBCD(K1)
   20 IF(BND(I,J).EQ.2) KOUT(J) = BLANK
```

BEST AVAILABLE COPY

```
21 PRINT 5, (KOUT(J), J=1,N,NSP)

   DO 23 I=1,M
   DO 23 J=1,N
   IF(SN(I,J).EQ.0) GO TO 22
   TA(I,J) = 0.0
   GO TO 23
22 TA(I,J) = -0.5/DX*(SN(I+1,J) - SN(I-1,J))
23 CONTINUE
   CALL BNDARY(M,N,M1,N1,IBF,JBF,NBPF,TA)

   P = SCALE(TA,M,N)
   PRINT 24, P
24 FORMAT(1H0, 37X, 19HV VELOCITY FUNCTION, 3X, 2HP=F9.2//)
   DO 25 I=1,M,NSP
   DO 25 J=1,N,NSP
   K1 = P*TA(I,J) + .5
   KOUT(J) = INBCD(K1)
25 IF(SN(I,J).EQ.2) KOUT(J) = BLANK
26 PRINT 5, (KOUT(J), J=1,N,NSP)

   DO 28 I=1,M
   DO 28 J=1,N
   IF(SN(I,J).EQ.0) GO TO 27
   TA(I,J) = 0.0
   GO TO 28
27 T1 = -.5/DX*(SN(I,J+1) - SN(I,J-1))
   T2 =  .5/DX*(SN(I+1,J) - SN(I-1,J))
   TA(I,J) = SQRT(T1*T1 + T2*T2)
28 CONTINUE
   CALL BNDARY(M,N,M1,N1,IBF,JBF,NBPF,TA)

   P = SCALE(TA,M,N)
   PRINT 29, P
29 FORMAT(1H0, 37X, 21HTOTAL VELOCITY VECTOR, 3X, 2HP=F9.2//)
   DO 31 I=1,M,NSP
   DO 30 J=1,N,NSP
   K1 = P*TA(I,J) + .5
   KOUT(J) = INBCD(K1)
30 IF(SN(I,J).EQ.2) KOUT(J) = BLANK
31 PRINT 5, (KOUT(J), J=1,N,NSP)

32 RETURN


   SUBROUTINE BNDARY(M,N,M1,N1,IBF,JBF,NBPF,TA)
   DIMENSION IBF(100,4), JBF(100,4), NBPF(6), TA(61,26)

   DO 5 KK=1,4
   L = NBPF(KK)
   IF(L.EQ.0) GO TO 5
   DO 4 K=1,L
   I = IBF(K,KK)
   J = JBF(K,KK)
   IF(J.NE.1) GO TO 1
   TA(I,1) = TA(I,2)
   GO TO 4
 1 IF(J.NE.N) GO TO 2
   TA(I,N) = TA(I,N1)
```

BEST AVAILABLE COPY

```
      GO TO 4
    2 IF(I.NE.1) GO TO 3
      TA(1,J) = TA(2,J)
      GO TO 4
    3 IF(I.NE.M) GO TO 4
      TA(M,J) = TA(M1,J)
    4 CONTINUE
    5 CONTINUE
      RETURN
      END
```

```
$I FOR VBND1,VBND1
      SUBROUTINE VBND1(IVS,IVB,JVB,KVB)
      DIMENSION IVS(200), IVB(200), JVB(200), BND(61,26)
      DIMENSION ISB(200), JSB(200), JSE(200)
      DIMENSION ISB1(200), JSB1(200), ISE1(200)
      COMMON/COM3/ ISB, JSB, JSE, KSB, ISB1, ISE1, JSB1, KSB1
      INTEGER BND
      COMMON/COM1/ BND, M, N, M1, N1, M2, N2, M3, N3
C
      K = 0
      DO 6 I=2,M1
      DO 7 J=2,N1
C
C                                          BOUNDARY FACING LEFT
      IF(BND(I,J).NE.0 .OR. BND(I+1,J).NE.1) GO TO 1
      K = K + 1
      IVS(K) = 1
      GO TO 4
C                                          BOUNDARY FACING RIGHT
    1 IF(BND(I,J).NE.0 .OR. BND(I-1,J).NE.1) GO TO 2
      K = K + 1
      IVS(K) = 2
      GO TO 4
C                                          BOUNDARY FACING UP
    2 IF(BND(I,J).NE.0 .OR. BND(I,J-1).NE.1) GO TO 3
      K = K + 1
      IVS(K) = 3
      GO TO 4
C                                          BOUNDARY FACING DOWN
    3 IF(BND(I,J).NE.0 .OR. BND(I,J+1).NE.1) GO TO 7
      K = K + 1
      IVS(K) = 4
C
C
    4 IF(K.LT.200) GO TO 6
      PRINT 5
    5 FORMAT(1H1,///,30X, 32HBND COEF. DIM. HAS BEEN EXCEEDED )
      CALL EXIT
    6 IVB(K) = I
      JVB(K) = J
      KVB = K
    7 CONTINUE
    8 CONTINUE

C         COLUMN BOUNDS

      K = 0
      DO 13 I=2,M1
      JB = 2
    9 IS = 1
      DO 12 J=JB,N
      GO TO (10,11), IS
   10 IF(BND(I,J).GE.1) GO TO 12
      K = K + 1
      IF(K.GT.200) STOP VBND1
      ISB(K) = I
      JSB(K) = J
      IS = 2
      GO TO 12
```

BEST AVAILABLE COPY

```
   11 IF(BND(I,J).EQ.0) GO TO 12
      JSE(K) = J-1
      JB = J
      GO TO 9
   12 CONTINUE
   13 CONTINUE
      KSB = K

C          ROW BOUNDS

      K = 0
      DO 18 J=2,N1
      IB = 2
   14 JS = 1
      DO 17 I=IB,M
      GO TO (15,16), JS
   15 IF(BND(I,J).GE.1) GO TO 17
      K = K + 1
      IF(K.GT.200) STOP VBND1
      ISB1(K) = I
      JSB1(K) = J
      JS = 2
      GO TO 17
   16 IF(BND(I,J).EQ.0) GO TO 17
      ISE1(K) = I-1
      IB = I
      GO TO 14
   17 CONTINUE
   18 CONTINUE
      KSB1 = K

      RETURN
      END
```

```
C I FOR STREAM,STREAM
      SUBROUTINE STREAM(SN,VN,N2,ITER,ES,OMEGA,ENEW,JJ,
     1                  ISB,JSB,A,S1,S2,S3,S4,DOMEGA,OMEGA1,ESTR)
      DIMENSION SN(61,26), VN(61,26), ISB(200), JSB(200), JSE(200)
      DIMENSION A(61,26), S1(61,26), S2(61,26), S3(61,26), S4(61,26)
      DIMENSION ISB1(200), JSB1(200), ISE1(200)
      COMMON/CONS/ ISB, JSB, JSE, KSB, ISB1, ISE1, JSB1, KSB1


      OMEGA1 = OMEGA - FLOAT(JJ)*DOMEGA
      AA = 1.-OMEGA1
      ESUMT = 0.45 - FLOAT(JJ)*0.015
      IF(ESUMT.LT.0.10) ESUMT = 0.10
      DO 3 ITER=1,150,2
      ES = 0.0
      ENEW = ENEW
      ENEW = 0.0

C         ROW SWEEPS....

      DO 2 K=1,KSB1
      J = JSB1(K)
      IB = ISB1(K)
      IE = ISE1(K)

      DO 1 I=IB,IE
      AB = OMEGA1/(S1(I,J) + S2(I,J) + S3(I,J) + S4(I,J))
      SNEW = AA*SN(I,J) + AB*(S1(I,J)*SN(I+1,J) + S2(I,J)*SN(I-1,J) +
     1       S3(I,J)*SN(I,J+1) + S4(I,J)*SN(I,J-1) + H2*VN(I,J))
      DIF = ABS(SN(I,J)-SNEW)
      ES = AMAX1(ES, DIF)
      ENEW = ENEW + DIF
    1 SN(I,J) = SNEW
    2 CONTINUE

      IF(ES.LT.10.0*ESTR .AND. ENEW.LT.ESUMT) GO TO 5

    3 CONTINUE
      PRINT 4, ES, ENEW, OMEGA
    4 FORMAT(1H ,////, 20X, 21HSOR FAILS TO CONVERGE//20X, 3HES=,
     1  F20.6, 3X, 5HENEW=F20.6, 3X, 6HOMEGA=F10.6)
      CALL EXIT

    5 ESUMT = 0.60*ESUMT + 0.40*ENEW
      RETURN
      END
```

BEST AVAILABLE COPY

```
WI FOR VBND,VBND
      SUBROUTINE VBND(SN,VN,IVB,JVB,IVS,KVB,SFVBND,H2,A,SFVBD1,        229
     1                S1,S2,S3,S4)
      DIMENSION IVS(200), IVB(200), JVB(200)
      DIMENSION SN(61,26), VN(61,26), A(61,26), BND(61,26)
      DIMENSION S1(61,26), S2(61,26), S3(61,26), S4(61,26)
      DIMENSION SS(5)
C
      COMMON/COM1/ BND, M, N, M1, N1, M2, N2, M3, N3
C
      INTEGER BND
C
C
C
      DO 16 KK=1,KVB
      I = IVB(KK)
      J = JVB(KK)
      IS = IVS(KK)
      GO TO (1,5,9,13), IS
C
C
C       BOUNDARY FACING LEFT
C
    1 DO 4 K=1,3
      L = J+2-K
      IF(BND(I,L).EQ.1) GO TO 2
      II = I
      B = A(II+1,L)
      C = 1. - B
      GO TO 3
    2 II = I - 1
      B = A(II+1,L)
      C = 2. - B
    3 BC = B + C
      B1 = B + 1.
    4 SS(K) = BC*C*SN(II-1,L)/B1 - C*(1.+BC)*SN(II,L)/B
     1        + (1.+BC)*BC*SN(II+1,L)/(B*B1)
      ST = SS(2) + .5*H2*(-4.*VN(I,J) + 2.*VN(I-1,J) -
     1          (SS(1) - 2.*SS(2) + SS(3))/H2)
      SS(4) = SN(I-1,J+1)
      SS(5) = SN(I-1,J-1)
      GO TO 17
C
C       BOUNDARY FACING RIGHT
C
    5 DO 8 K=1,3
      L = J+2-K
      IF(BND(I,L).EQ.1) GO TO 6
      II = I
      B = A(II-1,L)
      C = 1.-B
      GO TO 7
    6 II = I + 1
      B = A(II-1,L)
      C = 2.-B
    7 BC = B + C
      B1 = B + 1.
    8 SS(K) = BC*C*SN(II+1,L)/B1 - C*(1.+BC)*SN(II,L)/B +
     1        (1.+BC)*BC*SN(II-1,L)/(B*B1)
```

BEST AVAILABLE COPY

```
      ST = SS(2) + .5*H2*(-4.*VN(I,J) + 2.*VN(I+1,J) -
     1          (SS(1) - 2.*SS(2) + SS(3))/H2)
      SS(4) = SN(I+1,J+1)
      SS(5) = SN(I+1,J-1)
      GO TO 17
C
C       BOUNDARY FACING UP
C
    9 DO 12 K=1,3
      L = I+2-K
      IF(BNE(L,J).EQ.1) GO TO 10
      JJ = J
      B = A(L,JJ-1)
      C = 1.-B
      GO TO 11
   10 JJ = J + 1
      B = A(L,JJ-1)
      C = 2.-B
   11 BC = B + C
      B1 = B + 1.
   12 SS(K) = BC*C*SN(L,JJ+1)/B1 - C*(1.+BC)*SN(L,JJ)/B +
     1          (1.+BC)*BC*SN(L,JJ-1)/(B*B1)
      ST = SS(2) + .5*H2*(-4.*VN(I,J) + 2.*VN(I,J+1) -
     1          (SS(1) - 2.*SS(2) + SS(3))/H2)
      SS(4) = SN(I-1,J+1)
      SS(5) = SN(I+1,J+1)
      GO TO 17
C
C       BOUNDARY FACING DOWN
C
   13 DO 16 K=1,3
      L = I+2-K
      IF(BNE(L,J).EQ.1) GO TO 14
      JJ = J
      B = A(L,JJ+1)
      C = 1.-B
      GO TO 15
   14 JJ = J + 1
      B = A(L,JJ+1)
      C = 2.-B
   15 BC = B + C
      B1 = B + 1.
   16 SS(K) = BC*C*SN(L,JJ-1)/B1 - C*(1.+BC)*SN(L,JJ)/B +
     1          (1.+BC)*BC*SN(L,JJ+1)/(B*B1)
      ST = SS(2) + .5*H2*(-4.*VN(I,J) + 2.*VN(I,J-1) -
     1          (SS(1) - 2.*SS(2) + SS(3))/H2)
      SS(4) = SN(I-1,J-1)
      SS(5) = SN(I+1,J-1)
C
   17 SN(I,J) = (1.0-SFVBD1)*ST + SFVBD1*SN(I,J)
      SS = A(I+1,J)*A(I,J+1)*A(I-1,J)*A(I,J-1)/(A(I+1,J)*A(I-1,J) +
     1          A(I,J+1)*A(I,J-1))
      TEM = 8.*(S1(I,J)*SN(I+1,J) + S2(I,J)*SN(I-1,J) +
     1      S3(I,J)*SN(I,J+1) + S4(I,J)*SN(I,J-1)) + SS(1) + SS(3) +
     2      SS(4) + SS(5) - 4.*(2./SS + 1.)*SN(I,J)
      VN(I,J) = -(1.-SFVBND)*TEM/(12.*H2) + SFVBND*VN(I,J)
   18 CONTINUE
C
C
```

BEST AVAILABLE COPY

```
C
      RETURN
      END
```

```
JI FOR VORT,VORT
      SUBROUTINE VORT(VN,VO,SN,SO,DT1,H,DH,SF,NIPR)
      DIMENSION SN(61,26), VN(61,26), BND(61,26), U(61), V(61)
      DIMENSION A(61,26),SO(61,26),VO(61,26),VT(26),TA(61,26)
      COMMON/CON1/ BND, M, N, M1, N1, M2, N2, M3, N3
      COMMON/CON2/ A
      INTEGER BND

      DO 10 NI=1,NIPR
C
C        ALTERNATING DIRECTION IMPLICIT METHOD....
C
      DO 8 J=3,N2
      IB = 2
    1 IS = 1
      DO 4 I=IB,M
      GO TO (2,3), IS
    2 IF(BND(I,J).GE.1 .OR. BND(I,J-1).GE.1 .OR. BND(I,J+1).GE.1
     1    .OR. BND(I-1,J).GE.1) GO TO 4
      IB = I
      IS = 2
      GO TO 4
    3 IF(BND(I,J).EQ.0 .AND. BND(I,J+1).EQ.0 .AND. BND(I,J-1).EQ.0
     1 .AND. BND(I+1,J).EQ.0) GO TO 4
      IE = I - 1
      GO TO 5
    4 CONTINUE
      GO TO 8
C
    5 IB1 = IB + 1
      IE1 = IE - 1
      U(IB-1) = 0.0
      V(IB-1) = 0.0
C     ROW OF FD POINTS....
      TA(IB-1,J) = .5*(VN(IB-1,J) + VO(IB-1,J))
      TA(IE+1,J) = .5*(VN(IE+1,J) + VO(IE+1,J))
C
      DO 6 I=IB,IE
      T1 = A(I+1,J) + A(I-1,J)
      T3 = A(I,J+1) + A(I,J-1)
      V1 = A(I-1,J)/(T1*A(I+1,J))
      V2 = (A(I+1,J) - A(I-1,J))/(A(I+1,J)*A(I-1,J))
      V3 = -A(I+1,J)/(T1*A(I-1,J))
      V4 = A(I,J-1)/(T3*A(I,J+1))
      V5 = (A(I,J+1)-A(I,J-1))/(A(I,J+1)*A(I,J-1))
      V6 = -A(I,J+1)/(T3*A(I,J-1))
      AA = DH*(V1*SO(I+1,J) + V2*SO(I,J) + V3*SO(I-1,J))
      BB = .5*DH*(V4*(SN(I,J+1)+SO(I,J+1)) + V5*(SN(I,J)-SO(I,J))
     1          + V6*(SN(I,J-1)+SO(I,J-1)))
      T2 = 2. + H*BB*A(I+1,J)
      T4 = 2. - H*AA*A(I,J+1)
      A1 = -T2/(T1*A(I-1,J))
      A3 = (H*BB*A(I-1,J) - 2.)/(T1*A(I+1,J))
      A2 = DT1 + (T2 - H*BB*A(I-1,J))/(A(I+1,J)*A(I-1,J))
      A5 = DT1 - (T4 + H*AA*A(I,J-1))/(A(I,J+1)*A(I,J-1))
      A4 = T4/(T3*A(I,J-1))
      A6 = (H*AA*A(I,J-1) + 2.)/(T3*A(I,J+1))
      C1 = -IB/I
      C2 = -I/IE
```

BEST AVAILABLE COPY

```
      D(I) = C1*A1*TA(I-1,J) + A4*VO(I,J-1) + A5*VO(I,J) +
     1       A6*VO(I,J+1) + C2*A3*TA(I+1,J)
      TEM = A2 - A1*U(I-1)
      U(I) = A3/TEM
    6 V(I) = (D(I) - A1*V(I-1))/TEM
C
      TA(IE,J) = V(IE)
      DO 7 I=IB,IE1
      K = IE1 + IB - I
    7 TA(K,J) = V(K) - U(K)*TA(K+1,J)
      IB = IE + 2
      GO TO 1
    8 CONTINUE
C
      DO 17 I=3,M2
C
      JB = 3
    9 IS = 1
      DO 12 J=JB,N
      GO TO (10,11), IS
   10 IF(BND(I,J).GE.1 .OR. BND(I-1,J).GE.1 .OR.BND(I+1,J).GE.1
     1    .OR. BND(I,J-1).GE.1) GO TO 12
      JB = J
      IS = 2
      GO TO 12
   11 IF(BND(I,J).EQ.0 .AND. BND(I-1,J).EQ.0 .AND. BND(I+1,J).EQ.0
     1 .AND. BND(I,J+1).EQ.0) GO TO 12
      JE = J - 1
      GO TO 13
   12 CONTINUE
      GO TO 17
C
   13 JB1 = JB + 1
      JE1 = JE - 1
      U(JB-1) = 0.0
      V(JB-1) = 0.0
C     COLUMN END POINTS.....
      TA(I,JB-1) = .5*(VN(I,JB-1) + VO(I,JB-1))
      TA(I,JE+1) = .5*(VN(I,JE+1) + VO(I,JE+1))
C
      DO 14 J=JB,JE
      T1 = A(I+1,J) + A(I-1,J)
      T3 = A(I,J+1) + A(I,J-1)
      V1 = A(I-1,J)/(T1*A(I+1,J))
      V2 = (A(I+1,J) - A(I-1,J))/(A(I+1,J)*A(I-1,J))
      V3 = -A(I+1,J)/(T1*A(I-1,J))
      V4 = A(I,J-1)/(T3*A(I,J+1))
      V5 = (A(I,J+1)-A(I,J-1))/(A(I,J+1)*A(I,J-1))
      V6 = -A(I,J+1)/(T3*A(I,J-1))
      AA = DH*(V1*SN(I+1,J) + V2*SN(I,J) + V3*SN(I-1,J))
      BB = .5*DH*(V4*(SN(I,J+1)+SO(I,J+1)) + V5*(SN(I,J)-SO(I,J))
     1          + V6*(SN(I,J-1)+SO(I,J-1)))
      T2 = 2. - H*AA*A(I,J+1)
      T4 = 2. - H*BB*A(I-1,J)
      A1 =-T2/(T3*A(I,J-1))
      A3 = -(H*AA*A(I,J-1) + 2.)/(T3*A(I,J+1))
      A2 = DT1 + (T2 + H*AA*A(I,J-1))/(A(I,J+1)*A(I,J-1))
      A5 = DT1 - (T4 + H*BB*A(I+1,J))/(A(I+1,J)*A(I-1,J))
      A4 = (H*BB*A(I+1,J) + 2.)/(T1*A(I-1,J))
```

```
      A6 = T4/(T1*A(I+1,J))
      C1 = -JB/J
      C2 = -J/JE
      D(J) = C1*A1*VN(I,J-1) + A4*TA(I-1,J) + A5*TA(I,J) +
     1       A6*TA(I+1,J) + C2*A3*VN(I,J+1)
      TEM = A2 - A1*U(J-1)
      U(J) = A3/TEM
      T1 = V(J-1)/TEM
      T2 = D(J)/TEM
   14 V(J) = T2 - A1*T1
C
      VT(JE) = V(JE)
      DO 15 J=JB,JE1
      K = JE1 + JB - J
   15 VT(K) = V(K) - U(K)*VT(K+1)
C
      DO 16 J=JB,JE
   16 VN(I,J) = (1.-SF)*VT(J) + SF*VN(I,J)
C
      JB = JE + 2
      GO TO 9
   17 CONTINUE

   18 CONTINUE
C
      END
```

BEST AVAILABLE COPY

```
$I FOR LABLE,LABLE
      SUBROUTINE LABLE(X,FMT)                                          235
      DIMENSION FMT(8), IBUF(22)
      DIMENSION FL(600)
      INTEGER FL
      COMMON/INTRC1/ FL, IX, IY, IA
      CALL ENCODE(IBUF)
      WRITE(23,FMT) X
      IY = IY - 32
      CALL LNTYPE(3)
      CALL BLANK(1)
      CALL LINE(FL,IX,IY,IA)
      CALL BLANK(0)
      CALL CHAR(FL,IBUF,IA)
      RETURN
      END
```

```
UN ASM INBCD,INBCD
5(1)      REGNAM
INBCD*    SA          A1,T
          SA          A2,T+1
          SA          A3,T+2
          SA          A4,T+3
          LA          A0,*0,B11
          LA          A4,BLANK
          JP          A0,5+3
          LA          A4,BLAMIN
          LN          A0,A0
          LA,14       A3,12
          USL         A0,36
          DI,14       A0,10
          AA,14       A1,060
          USL         A1,6
          AA,14       A3,6
          JNZ         A0,5-5
          LA          A0,A4
          LUSL        A1,36
          SA,1        A3,5+1
          LUSL        A0,0
          OR,14       A0,0404
          LA          A0,A1
          LA          A1,T
          LA          A2,T+1
          LA          A3,T+2
          LA          A4,T+3
          J           2,B11
BLAMIN    '      -'
BLANK     ' '
5(0),T    RES         4
          END
```

```
$I FOR INTRUP,INTRUP
      SUBROUTINE INTRUP                                                    237
      LOGICAL INTR, RESTRT, REDO, EPLOT, SWAPON
      COMMON/INTRC/ OMEGA, R, NFRAME, SFVBND, SFVORT, BPR1, BPR2,
     1              BPR3, BPR4, DISPLY, DR, DT, EVOR, ESTR, IWRITE,
     2              NIPR, RESTRT, REDO, EPLOT, SWAPON
      COMMON/TEL/ IPOINT, NCH, INS(22)
      COMMON/INTR1/ NPLOT
      INTEGER OK

      INTR = .FALSE.
      CALL IDLE
      CALL CHRINT(1,$1)
      IF(INTR) CALL TYPLIN('^A')
      RETURN


    1 INTR = .TRUE.
    2 DO 3 K=1,72
      CALL GETCHR(ICH)
      INS(K) = ICH
      IF(ICH.EQ.'␍' .OR. ICH.EQ.' ') GO TO 4
    3 NCH = K

    4 CALL WORD(INS,IWORD)
      IF(IWORD.NE.'ABORT') GO TO 5
      CALL TYPLIN('␍␊EXIT^A')
      CALL EXIT

    5 IF(IWORD.NE.'SWAP') GO TO 6
      CALL SWPCHR('␍')
      SWAPON = .TRUE.
      CALL TYPLIN('␍OUT␍^A')
      CALL SWAP(0)
      CALL GETCHR(ICH)
      CALL TYPLIN('␍IN␍^A')
      CALL INTRET

    6 IF(IWORD.NE.'OMEGA') GO TO 7
      OMEGA = RNUM(0)
      CALL VARDIS
      GO TO 2
    7 IF(IWORD.NE.'R') GO TO 8
      R = RNUM(0)
      CALL VARDIS
      GO TO 2
    8 IF(IWORD.NE.'NFRAME') GO TO 9
      NFRAME = RNUM(0)
      CALL VARDIS
      GO TO 2
    9 IF(IWORD.NE.'SFVBND') GO TO 10
      SFVBND = RNUM(0)
      CALL VARDIS
      GO TO 2
   10 IF(IWORD.NE.'SFVORT') GO TO 11
      SFVORT = RNUM(0)
      CALL VARDIS
      GO TO 2
   11 IF(IWORD.NE.'BPR1') GO TO 12
```

```
      BPR1 = RNUM(0)
      CALL VARDIS
      GO TO 2
   12 IF(IWORD.NE.'BPR2') GO TO 13
      BPR2 = RNUM(0)
      CALL VARDIS
      GO TO 2
   13 IF(IWORD.NE.'BPR3') GO TO 14
      BPR3 = RNUM(0)
      CALL VARDIS
      GO TO 2
   14 IF(IWORD.NE.'BPR4') GO TO 15
      BPR4 = RNUM(0)
      CALL VARDIS
      GO TO 2
   15 IF(IWORD.NE.'DISPLY') GO TO 16
      DISPLY = RNUM(0)
      CALL VARDIS
      GO TO 2
   16 IF(IWORD.NE.'ER') GO TO 17
      ER = RNUM(0)
      CALL VARDIS
      GO TO 2
   17 IF(IWORD.NE.'DT') GO TO 18
      DT = RNUM(0)
      CALL VARDIS
      GO TO 2
   18 IF(IWORD.NE.'EVOR') GO TO 19
      EVOR = RNUM(0)
      CALL VARDIS
      GO TO 2
   19 IF(IWORD.NE.'ESTR') GO TO 20
      ESTR = RNUM(0)
      CALL VARDIS
      GO TO 2
   20 IF(IWORD.NE.'IWRITE') GO TO 21
      IWRITE = RNUM(0)
      CALL VARDIS
      GO TO 2
   21 IF(IWORD.NE.'NIPR') GO TO 22
      NIPR = RNUM(0)
      CALL VARDIS
      GO TO 2
   22 IF(IWORD.NE.'PLOT') GO TO 24
      CALL LOCUSD(FL,IA)
      NPLOT = NPLOT + 1
      IF(NPLOT.EQ.1) CALL DELALL
      CALL DRAWRT(FL,IA,NPLOT,OK)
      IF(OK.EQ.0) CALL TYPLIN('WARNING... OK=0Δ')
      CALL ENCODE(INS)
      WRITE(23,23) IA
   23 FORMAT(I4, 'Δ')
      CALL TYPLIN('#PLOT OF Δ')
      CALL TYPLIN(INS)
      CALL TYPLIN(' WORDS#Δ')
      CALL INTRET
   24 IF(IWORD.NE.'RESTRT') GO TO 25
      RESTRT = .TRUE.
      CALL INTRET
```

BEST AVAILABLE COPY

```
   25 IF(IWORD.NE.'REDO') GO TO 26
      REDO = .TRUE.
      CALL INTRET
   26 IF(IWORD.NE.'EPLOT') GO TO 29
      ISW = RNUM(TEM) + 1.0
      GO TO (27,28), ISW
   27 EPLOT = .FALSE.
      CALL INTRET
   28 EPLOT = .TRUE.
      CALL INTRET
   29 IF(IWORD.NE.'CONT') GO TO 30
      CALL INTRET
   30 IF(IWORD.NE.'VAR') GO TO 31
      CALL VARDIS
      GO TO 2
   31 CALL TYPLIN('n?␣')
      GO TO 2


      SUBROUTINE WORD(INS1,IWORD)
      DIMENSION INS1(22)
      COMMON/TEL/ IPOINT, NCH, INS(22)
      IWORD = '      '
      K = 0
      DO 1 I=1,36,6
      K = K + 1
      IF(INS1(K).NE.'=' .AND. INS1(K).NE.'n') GO TO 1
      IPOINT = K + 1
      RETURN
    1 FLD(1-1,6,IWORD) = FLD(0,6,INS1(K))
      IPOINT = 0
      RETURN
C
      FUNCTION RNUM(TEM)
      REAL NUM(20)
      COMMON/TEL/ IPOINT, NCH, INS(22)

      ISW = 1
      ID = 0
      II = 0
      DO 4 I=IPOINT,NCH
      IF(INS(I).NE.',') GO TO 1
      ISW = 2
      GO TO 4
    1 II = II + 1
      GO TO (3,2), ISW
    2 ID = ID + 1
    3 NUM(II) = (INS(I)-'0')/2**30
    4 CONTINUE

      NUM1 = 0
      DO 5 K=1,II
      KK = II + 1 - K
    5 NUM1 = NUM1 + NUM(KK)*10**(K-1)

      RNUM = FLOAT(NUM1)/FLOAT(10**ID)
      RETURN
```

```
$I FOR ERRDIS,ERRDIS
      SUBROUTINE ERRDIS(SFRAME,ITER,ITERS,EV,ES,R,EVOR,ESTR,M,N,BLN,
     1                  XBP,YBP,NFILE,TTIME,TVBND,TVORT,TSTR,TMISC,TA,
     2                  TTBND,TTMISC,TTVORT,TTVBND,TTSTR,NFL)
      DIMENSION MSG1(5), MSG2(3), MSG3(3), MSG4(8), TA(61,26),
     1          BLN(4), XBP(100,4), YBP(100,4), II(100), JJ(100),
     2          XTEM(100), YTEM(100), ES(60), EV(60), ITERS(60)
C
      COMMON/ER1/ FL(1500), IA
      COMMON/ER2/ IAS
      INTEGER R
      COMMON/CONS/ XTEM, YTEM
      DATA MSG1/2HEA, 2HRA, 2HNA, 2HOA, 2HRA/
      DATA MSG2/2HSA, 2HOA, 2HKA, 2H A, 2HIA, 2HTA, 2HEA, 2HRA/
      DATA MSG3/3H20A, 3H40A, 3H60A/
      DATA MSG4/6H.0001A, 6H .001A, 6H  .01A, 6H   .1A,
     1          6H    1.A, 6H   10.A, 6H 100.A, 6H1000.A/


      IF(ITER.GT.1) GO TO 10
      CALL IDI(FL,IA)
      CALL INTEN(2)

      CALL ENTYPE(0)
      CALL LINE(FL,0,0,IA)
      CALL ENTYPE(3)

      CALL LN(110,900,3)
      CALL LN(110,600,2)
      CALL LN(350,600,2)

      CALL LN(430,900,3)
      CALL LN(430,600,2)
      CALL LN(670,600,2)

      CALL LN(750,900,3)
      CALL LN(750,600,2)
      CALL LN(990,600,2)

      CALL LABEL(250,500,16HSTREAM FUNCTION$)
      CALL LABEL(685,500,19HVORTICITY FUNCTION$)
      CALL LABEL(160,550,10HITERATION$)
      CALL LABEL(460,550,10HITERATION$)
      CALL LABEL(760,550,10HITERATION$)
      DO 1 I=1,5
      CALL LABEL(30,820-I*20, MSG1(I))
    1 CALL LABEL(650,820-I*20, MSG1(I))
      DO 2 I=1,8
    2 CALL LABEL(355,850-I*20, MSG2(I))

      CALL OFFSET(3)
      DO 3 J=1,3
      DO 3 I=1,3
      I1 = 110 + 80*I + (J-1)*500
      CALL LN(I1,600,3)
      CALL LN(I1,590,2)
      CALL LABEL(I1-4,575,MSG3(I))
    3 CONTINUE
```

```
      CALL OFFSET(0)
      CALL LABEL(220,132,32HLOCATION OF LARGEST ERROR TERMSA)      241
      CALL OFFSET(3)

      R1 = 300./FLOAT(ITERS(1)+5)
      DO 4 I=1,6
      K = 600 + (I-1)*60
      CALL LN(750,K,3)
      CALL LN(740,K,2)
    4 CALL LABEL(680,K-4,MSG4(I+1))
      DO 5 I=1,5
      I1 = 600+I*60
      CALL LN(430,I1,3)
      CALL LN(420,I1,2)
    5 CALL NUMBER(365,I1-4,ITERS(1)*I/5)
      DO 6 I=1,8
      K = 600. + FLOAT(I-1)*42.85
      CALL LN(110,K,3)
      CALL LN(100,K,2)
    6 CALL LABEL(40,K-4,MSG4(I))
      CALL OFFSET(0)

C         DISPLAY BOUNDARY POINTS

      DO 9 K=1,NFILE
      K1 = NLN(K)
      DO 7 I=1,K1
      XTEM(I) = XBP(I,K) - 1.
    7 YTEM(I) = YBP(I,K) - 1.
      LL = NLN(K)
      CALL LNPLT(XTEM,YTEM,LL)
      I1 = XTEM(1)
      J1 = YTEM(1)
      CALL LN(I1*10+200,J1*10+200,3)
      DO 8 L=2,LL
      I1 = XTEM(L)
      J1 = YTEM(L)
    8 CALL LN(I1*10+200,J1*10+200,2)
    9 CONTINUE
      CALL SNDFLE(FL,IA)
      IAS = IA

C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

   10 IA = IAS
      CALL FLPINS(FL,IA)
      CALL LNTYPE(0)
      CALL LINE(FL,0,0,IA)
      CALL LNTYPE(3)
      N1 = 110 + 4*ITER
      I = 600. + (ALOG10(ES(ITER))+4.)*50.
      CALL LN(N1,600,3)
      CALL LN(N1,I,2)

      I = 600. + R1*FLOAT(ITERS(ITER))
      CALL LN(N1+320,600,3)
      CALL LN(N1+320,I,2)

      I = 600. + (ALOG10(EV(ITER))+3.)*50.
```

```
      CALL LN(N1+640,600,3)
      CALL LN(N1+640,1,2)

      IA5 = 1A
      CALL OFFSET(5)
      I = 600. + (ALOG10(ESTR)+4.)*50.
      CALL LN(110,I,3)
      CALL LN(350,I,2)
      CALL LABEL(253,I+16,6HERROR△)
      CALL LABEL(230,I+4,10HTOLERANCE△)

      I = 600. + (ALOG10(EVOR)+3.)*50.
      CALL LN(750,I,3)
      CALL LN(990,I,2)
      CALL LABEL(893,I+16,6HERROR△)
      CALL LABEL(870,I+4,10HTOLERANCE△)
      CALL OFFSET(0)

C        FIND LOCATION OF LARGEST ERROR TERMS

      TMAX = 0.0
      KN = 0
      DO 11 I=1,M
      DO 11 J=1,N
   11 TMAX = AMAX1(TMAX,ABS(TA(I,J)))
      TOL = .20*TMAX
      DO 12 I=1,M
      DO 12 J=1,N
      IF(ABS(TA(I,J)).LT.TOL) GO TO 12
      KN = KN + 1
      IF(KN.GE.100) GO TO 13
      II(KN) = I
      JJ(KN) = J
   12 CONTINUE

C        POSITION A DOT AT THE LOCATION OF THE ERROR TERMS

   13 KN = KN - 1
      CALL ENTYPE(1)
      DO 14 K=1,KN
      I = (II(K)-1)*10+200
      J = (JJ(K)-1)*10+200
   14 CALL LINE(FL,I,J,IA)

      CALL OFFSET(3)
      CALL LABEL(130,80,NFRAME,'(11HFRAME NO.= ,I2,1H△)')
      CALL LABEL(130,60,K,'(17HREYNOLDS NUMBER= ,I5,1H△)')
      CALL LABEL(130,40,TTIME,'(20HTOTAL ELAPSED TIME= ,F4.2,5H MIN△)')
      CALL LABEL(130,20,TTCONV,'(18HTIME TO CONVERGE= ,F5.2,5H SEC△)')

      CALL LABEL(530,80,TVBND,'(20HVORTICITY BOUNDARY= ,F4.2,5H SEC△)')
      CALL LABEL(530,60,TVORT,'(20HVORTICITY INTERIOR= ,F4.2,5H SEC△)')
      CALL LABEL(554,40,TSTR,'(17HSTREAM INTERIOR= ,F4.2,5H SEC△)')
      CALL LABEL(578,20,TMISC,'(14HMISCELANEOUS= ,F4.2,5H SEC△)')

      CALL LABEL(840,100,'TOTAL(SEC)△')
      CALL NUMBR(850,80,TTVBND)
      CALL NUMBR(850,60,TTVORT)
      CALL NUMBR(850,40,TTSTR)
```

```
      CALL NUMBR(650,20,TIMISC)
      CALL OFFSET(0)                                                    243

      CALL SNDFLE(FL,IA)
      CALL OFFSET(0)
      RETURN

C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      SUBROUTINE LN(IX,IY,IS1)
      COMMON/ER1/ FL(1586), IA
      INTEGER FL
      GO TO (3,2,1), IS1
    1 CALL BLANK(1)
      CALL LINE(FL,IX,IY,IA)
      CALL BLANK(0)
      RETURN
    2 CALL LINE(FL,IX,IY,IA)
      RETURN
    3 CALL LNTYPE(1)
      CALL LINE(FL,IX,IY,IA)
      RETURN

      SUBROUTINE LABEL(IX,IY,ICH)
      COMMON/ER1/ FL(1586), IA
      INTEGER FL
      CALL BLANK(1)
      CALL LINE(FL,IX,IY,IA)
      CALL BLANK(0)
      CALL CHAR(FL,ICH,IA)
      RETURN

      SUBROUTINE NUMBER(IX,IY,NUM)
      CALL ENCODE(IBUF)
      WRITE(23,1) NUM
    1 FORMAT(I4, 1HA)
      CALL LABEL(IX,IY,IBUF)
      RETURN

      SUBROUTINE NUMBR(IX,IY,TEM)
      COMMON/ER1/ FL(1586), IA
      INTEGER FL
      CALL ENCODE(IBUF)
      WRITE(23,1) TEM
    1 FORMAT(F5.2, 1HA)
      CALL OFFSET(3)
      CALL LABEL(IX,IY,IBUF)
      RETURN

      SUBROUTINE LABLE(IX,IY,X,FMT)
      DIMENSION FMT(8), IBUF(22)
      COMMON/ER1/ FL(1586), IA
      INTEGER FL

      CALL ENCODE(IBUF)
      WRITE(23,FMT) X
      CALL BLANK(1)
      CALL LINE(FL,IX,IY,IA)
      CALL BLANK(0)
      CALL CHAR(FL,IBUF,IA)
```

```
RETURN
END
```

```
*I FOR SCALE,SCALE
      FUNCTION SCALE(F,M,N)
      DIMENSION F(61,25)
C
C       THIS FUNCTION ROUTINE CALCULATES A MULTIPLICATION
C       FACTOR, A POWER OF 10, TO BE USED IN THE PRINTOUT
C
      BIG = 0.0
      DO 1 I=1,M
      DO 1 J=1,N
    1 BIG = AMAX1(BIG, ABS(F(I,J)))
      IF(BIG.GT.1.E-8) GO TO 2
      SCALE = 0.0
      RETURN
    2 L = INT(ALOG10(BIG))
C       I.E., 10**L = BIG
      IF(ISIGN(1,L).GT.0) GO TO 3
      SCALE = 10.0**(3-L)
      GO TO 4
    3 SCALE = 10.0**(3-L-1)
    4 RETURN
      END
```

```
JI FOR DATA,DATA
      DIMENSION XBP(100,4), YBP(100,4), BLN(4), A(61,26), BND(61,26),
     1          BPF(100,4), BPS(100,4), IBF(100,4), JBF(100,4),
     2          IBS(100,4), JBS(100,4), NBPF(6), NBPS(6), IFP(2)
      INTEGER BND, BLANK, PROFIL
      LOGICAL INTIOR

      DATA BLANK/6H      /
      DATA PROFIL/4HPARA/
C
C
      READ 1, IFP(1), IFP(2)
    1 FORMAT(15X, A4, 15X, A4)
C
      EPS1 = .21
      EPS2 = 1. - EPS1
C
      DO 2 I=1,61
      DO 2 J=1,31
      DO 2 K=1,4
    2 A(I,J) = 1.
C
C
      CALL DATA1(NFILE,XBP,YBP,BLN)
C
      PRINT 3
    3 FORMAT(1H1, 50X, 15HBOUNDARY POINTS//25X, 4(11X, 1HX,  8X, 1HY))
      KK = 0
      DO 4 K=1,NFILE
    4 KK = MAX0(KK, INT(BLN(K)) )
      DO 5 I=1,KK
    5 PRINT 6, I, (XBP(I,K), YBP(I,K), K=1,NFILE)
    6 FORMAT(1H , 24X, I2, 4(6X, F6.2, 3X, F6.2))
C
      M = 0
      N = 0
      DO 7 K=1,NFILE
      MM = BLN(K)
      DO 7 L=1,MM
      M = MAX0(M, IFIX(XBP(L,K)))
    7 N = MAX0(N, IFIX(YBP(L,K)))
      M1 = M - 1
      N1 = N - 1
C
C      CALCULATE BND ARRAY
C
C      TEST FOR FLUID BOUNDARIES ON BOTTOM AND TOP ROW
C
      IN1 = 1000
      I11 = 1000
      IN2 = 0
      I12 = 0
      DO 9 K=1,NFILE
      L = BLN(K)
      DO 9 KK=1,L
      I = XBP(KK,K)
      J = YBP(KK,K)
      IF(ABS(FLOAT(J)-1.0).LT.0.001) GO TO 8
      IF(ABS(FLOAT(J)-FLOAT(N)).GT.0.001) GO TO 9
```

```
      IN1 = MIN0(IN1,I)
      IN2 = MAX0(IN2,I)
      GO TO 9
    8 I11 = MIN0(I11,I)
      I12 = MAX0(I12,I)
    9 CONTINUE
      DO 10 I=I11,I12
   10 BND(I,1) = 1
      DO 11 I=IN1,IN2
   11 BND(I,N) = 1
C
C
C     IF BOUNDARY POINT IS WITHIN EPS1 OF NODE THEN NODE IS
C     TYPE 1 (BOUNDARY).....
C
      DO 15 K=1,NFILE
      L = BLN(K)
      DO 15 KK=1,L
      I = XBP(KK,K)
      J = YBP(KK,K)
      X1 = ABS(FLOAT(I)-XBP(KK,K))
      Y1 = ABS(FLOAT(J)-YBP(KK,K))
      IF(Y1.GT.0) GO TO 13
      IF(X1.GT.EPS1) GO TO 12
      BND(I,J) = 1
      GO TO 15
   12 IF(X1.LT.EPS2) GO TO 15
      BND(I+1,J) = 1
      GO TO 15
   13 IF(Y1.GT.EPS1) GO TO 14
      BND(I,J) = 1
      GO TO 15
   14 IF(Y1.LT.EPS2) GO TO 15
      BND(I,J+1) = 1
   15 CONTINUE
C
C     VERTICAL SCAN
C
      DO 25 K=1,NFILE
      L = BLN(K)
      DO 25 KK=1,L
      I = XBP(KK,K)
      J = YBP(KK,K)
      IF(BND(I,J).EQ.1) GO TO 25
C
      X1 = ABS(XBP(KK,K)-FLOAT(I))
      Y1 = ABS(YBP(KK,K)-FLOAT(J))
C
C     GO TO 69 IF XBP IS NOT ON A VERTICAL LINE
      IF(X1.GT.0.0) GO TO 20
      IF(Y1.GT.EPS1) GO TO 16
      BND(I,J) = 1
      GO TO 25
C
C     TEST FOR BOUNDARY POINT BELOW
   16 DO 19 K1=1,L
      IF(K1.EQ.KK) GO TO 19
      IF(XBP(K1,K)-XBP(KK,K)) 19, 17, 19
   17 IF(YBP(K1,K)-YBP(KK,K)) 18, 19, 19
```

```
   18 BND(I,J) = 1
      GO TO 25
   19 CONTINUE
      BND(I,J+1) = 1
      GO TO 25
C
   20 IF(X1.GT.EPS1) GO TO 21
      BND(I,J) = 1
      GO TO 25
C
C        TEST FOR BOUNDARY POINT TO THE LEFT
   21 DO 24 K1=1,L
      IF(YBP(K1,K)-YBP(KK,K)) 24, 22, 24
   22 IF(XBP(K1,K)-XBP(KK,K)) 23, 24, 24
   23 BND(I,J) = 1
      GO TO 25
   24 CONTINUE
      BND(I+1,J) = 1
C
   25 CONTINUE

C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

C          FILL IN REMAINING POINTS

      DO 33 I=1,M
      ISW = 1
      ISW1 = 1
      DO 26 J=1,N
   26 IF(BND(I,J).EQ.1) ISW = -ISW
      IF(ISW.LT.0) ISW1 = 2
C
      ISW = 1
      DO 32 J=1,N
      IF(ISW.LT.0) GO TO 27
      IF(BND(I,J).EQ.1) GO TO 30
      BND(I,J) = 2
      GO TO 32
   27 IF(BND(I,J).EQ.1) GO TO 28
      BND(I,J) = 0
      GO TO 32
   28 GO TO (30,29), ISW1
   29 ISW1 = 1
      GO TO 32
   30 IF(J.EQ.1) GO TO 31
      IF(BND(I,J-1).EQ.1) GO TO 32
   31 ISW = -ISW
   32 CONTINUE
   33 CONTINUE
C
      DO 34 J=2,N1
      IF(BND(1,J).EQ.0) BND(1,J) = 1
   34 IF(BND(M,J).EQ.0) BND(M,J)= 1
      DO 35 I=2,M1
      IF(BND(I,1).NE.1) BND(I,1) = 2
   35 IF(BND(I,N).NE.1) BND(I,N) = 2

C          THEN RESCAN TO TYPE THE ROW POINTS....
```

BEST AVAILABLE COPY

```
      DO 40 J=2,M1
      INTIOR = .FALSE.
      DO 39 I=1,M1
      IF(INTIOR) GO TO 37
      IF(BND(I,J).EQ.1) GO TO 36
      BND(I,J) = 2
      GO TO 39
   36 IF(BND(I+1,J).NE.0) GO TO 39
      INTIOR = .TRUE.
      GO TO 39
   37 IF(BND(I,J).EQ.1) GO TO 38
      BND(I,J) = 0
      GO TO 39
   38 INTIOR = .FALSE.
   39 CONTINUE
   40 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C       CALCULATE MESH DISTANCE A......
C
      DO 64 KK=1,NFILE
      L = BLN(KK)
      DO 64 K=1,L
      I = XBP(K,KK)
      J = YBP(K,KK)
      X1 = XBP(K,KK) - FLOAT(I)
      Y1 = YBP(K,KK) - FLOAT(J)
      IF(X1.GT.0.000001) GO TO 52
C
C       VERTICAL LINE
C
      IF(Y1.LT.0.0005 .OR. Y1.GT.0.9999) GO TO 64
      CALL BNDPT(IBP,XBP,YBP,I,K,KK,L)
      IF(Y1.LT.EPS1) GO TO 44
      IF(Y1.GT.EPS2) GO TO 50
      GO TO (42,43,41), IBP
   41 A(I,J+1) = Y1
   42 A(I,J) = 1.-Y1
      IF(BND(I,J+1).EQ.1) A(I,J+1) = 2.-Y1
      GO TO 64
   43 A(I,J+1) = Y1
      IF(BND(I,J).EQ.1) A(I,J) = 1.+Y1
      GO TO 64
C
   44 GO TO (46,47,45), IBP
   45 A(I,J+1) = Y1
   46 A(I,J) = 1.-Y1
      GO TO 64
   47 A(I,J) = 1.+Y1
      GO TO 64

   48 A(I,J+1) = 2.-Y1
   49 A(I,J+1) = Y1
   50 GO TO (48,51,49), IBP
      GO TO 64
   51 A(I,J+1) = Y1
      GO TO 64
C
```

```
C       HORIZONTAL LINE
C
   52 CALL BNDPT(IBP,YBP,XBP,J,K,KK,L)
      IF(X1.LT.EPS1) GO TO 55
      IF(X1.GT.EPS2) GO TO 60
      GO TO (54,55,53), IBP
   53 A(I+1,J) = X1
   54 A(I,J) = 1.-X1
      IF(BND(I+1,J).EQ.1) A(I,J)=2.-X1
      GO TO 64
   55 A(I+1,J) = X1
      IF(BND(I,J).EQ.1) A(I,J)=1.+X1
      GO TO 64
C
   56 GO TO (58,59,57), IBP
   57 A(I,J) = 1.+X1
   58 A(I,J) = 1.-X1
      GO TO 64
   59 A(I,J) = 1.+X1
      GO TO 64
C
   60 GO TO (62,63,61), IBP
   61 A(I+1,J) = X1
   62 A(I+1,J) = 2.-X1
      GO TO 64
   63 A(I+1,J)=X1
C
   64 CONTINUE
C
   65 PRINT 66
   66 FORMAT(1H1, 49X, 8HA VALUES//)
      DO 68 I=1,M
      DO 68 J=1,N
      IF(ABS(A(I,J)-1.0).LT.0.0001 .OR. BND(I,J).NE.0) GO TO 68
      PRINT 67, I, J, A(I,J)
   67 FORMAT(1H , 30X, 2HA(, I2, 1H,, I2, 2H)=, F6.3)
   68 CONTINUE
C
      PRINT 69
   69 FORMAT(1H1,/// 45X, 9HBND ARRAY //)
      DO 70 L=1,N
      J = N+1-L
   70 PRINT 71, J, (BND(I,J), I=1,M)
   71 FORMAT(1H , 16X, I2, 2X, 6111)
      PRINT 72, (I, I=10,M,10)
   72 FORMAT(1H0, 28X, I2, 8X, I2, 8X, I2, 8X, I2, 8X, I2, 8X, I2)

C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

C         FLUID BOUNDARY POINTS ON LEFT AND RIGHT BOUNDARIES.....

C
      I1 = 1
      I2 = 2
      IS = 0
   73 IS1 = 1
      K = 0
      DO 76 J=2,N
      GO TO (74,75), IS1
```

```
   74 IF(BNE(I2,J).GE.1) GO TO 75
      JA = J-1
      IS1 = 2
      GO TO 76
   75 IF(BNE(I2,J).EQ.0) GO TO 76
      JB = J
      GO TO 77
   76 CONTINUE

   77 TEM = JB - JA
      DO 80 J=JA,JB
      IF(IFF(IS+1).EQ.PROFIL) GO TO 78
      YT = FLOAT(J-JA)/TEM
      GO TO 79
   78 TEM1 = FLOAT(J-JA)/TEM
      YT = -2.*TEM1**3 + 3.*TEM1**2
   79 K = K + 1
      NBPF(IS+1) = K
      IBF(K,IS+1) = I1
      JBF(K,IS+1) = J
      BPF(K,IS+1) = YT
   80 CONTINUE
C
      IS = IS + 1
      GO TO (81,82), IS
   81 I1 = M
      I2 = M1
      GO TO 73

C        FLUID BOUNDARY POINTS ON UPPER AND LOWER BOUNDARIES....

   82 J1 = 1
      IS = 0
      IA = 0
      IB = 0
   83 IS1 = 1
      K = 0
      DO 85 I=2,M
      GO TO (84,85), IS1
   84 IF(BNE(I,J1).GT.1) GO TO 86
      IA = I
      IS1 = 2
      GO TO 86
   85 IF(BNE(I,J1).EQ.1) GO TO 86
      IB = I-1
      GO TO 87
   86 CONTINUE

   87 TEM = IB-IA
      IF(IFIX(TEM).LE.0) GO TO 89
      DO 88 I=IA,IB
      K = K + 1
      NBPF(IS+3) = K
      IBF(K,IS+3) = I
      JBF(K,IS+3) = J1
      BPF(K,IS+3) = 1.0
   88 CONTINUE

   89 IS = IS + 1
```

```
      GO TO (90,91), IS
   90 J1 = 1
      GO TO 83
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C        SOLID BOUNDARY POINTS.....
   91 DO 94 L=1,NFILE
      KK = BLN(L)
      IMAX = 0
      IMIN = 1000
      JMAX = 0
      JMIN = 1000
      DO 92 K=1,KK
      IMAX = MAXO(IMAX, IFIX(XBP(K,L)))
      IMIN = MINO(IMIN, IFIX(XBP(K,L)))
      JMAX = MAXO(JMAX, IFIX(YBP(K,L)))
   92 JMIN = MINO(JMIN, IFIX(YBP(K,L)))

      K = 0
      DO 93 I=IMIN,IMAX
      DO 93 J=JMIN,JMAX
      IF(BND(I,J).NE.1) GO TO 93
      K = K + 1
      NBPS(L) = K
      IF(K.GT.100) STOP ER1
      IBS(K,L) = I
      JBS(K,L) = J
      BPS(K,L) = 1.0
   93 CONTINUE
   94 CONTINUE
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      MAXF = 0
      MAXS = 0
      DO 95 K=1,6
      MAXF = MAXO(MAXF, NBPF(K))
   95 MAXS = MAXO(MAXS,NBPS(K))
      PRINT 96, ((IBF(K,L), JBF(K,L), BPF(K,L), L=1,4), K=1,MAXF)
   96 FORMAT(1H1, 30X, 'FLUID BOUNDARY POINTS'///(20X, 4(5X, 'B(', I2,
     1       ',', I2, ')=', F8.3)))
      PRINT 97, ((IBS(K,L), JBS(K,L), BPS(K,L), L=1,4), K=1,MAXS)
   97 FORMAT(1H1, 30X, 'SOLID BOUNDARY POINTS'///(20X, 4(5X, 'B(', I2,
     1       ',', I2, ')=', F8.3)))

      WRITE(8)  IFP, M, N, NFILE, XBP, YBP, BLN, BND, A,
     1          BPF, BPS, IBF, JBF, IBS, JBS, NBPF, NBPS

      CALL EXIT


      SUBROUTINE BNDPT(IBP,BPT1,BPT2,I2,K,KK,L)
      DIMENSION BPT1(200,4), BPT2(200,4)

C        THIS SUBROUTINE IS USED TO FIND WHETHER A GIVEN BOUNDARY
C        LINE LIES ON THE TOP/RIGHT OR BOTTOM/LEFT OF THE BOUNDARY
C        SURFACE.... IBP=1....TOP/RIGHT, IBP=2...BOTTOM/LEFT,
```

```
C        IBP=3...NEITHER
      DO 99 I1=1,L
      IF(I1.EQ.K .OR. ABS(GPT1(I1,KK)-FLOAT(I2)).GT.0.000001) GO TO 99
      IF(GPT2(I1,KK).LT.GPT2(K,KK)) GO TO 98
      IBP=2
      GO TO 100
   98 IBP=1
      GO TO 100
   99 CONTINUE
      IBP=3
  100 RETURN
      END
```

```
FOR DATA1,DATA1
      SUBROUTINE DATA1(NFILE,XBP,YBP,BLN)
      DIMENSION XBP(100,4), YBP(100,4), BLN(4)
      DIMENSION X(13), Y(13)
      DATA X/11.0, 11.0, 11.0, 12.0, 13.0, 14.0, 15.0, 15.0, 15.0,
     *       14.0, 13.0, 12.0, 11.0/
      DATA Y/10.0, 11.0, 12.0, 13.0, 13.0, 13.0, 12.0, 11.0, 10.0,
     *       9.0, 9.0, 9.0, 10.0/

      NFILE = 3
      DO 1 K=1,61
      XBP(K,1) = K
      XBP(K,2) = K
      YBP(K,1) = 1.0
    1 YBP(K,2) = 21.0
      BLN(1) = 61.0
      BLN(2) = 61.0
      DO 2 K=1,13
      XBP(K,3) = X(K)
    2 YBP(K,3) = Y(K)
      BLN(3) = 13

      RETURN
      END
```

```
*I FOR DISPLY,DISPLY
      DIMENSION BND(61,26), A(61,26), SN(61,26), VN(61,26), FL(3000)
      DIMENSION NPLT(80)                                                    255
      COMMON/MAIN/BND, A
      COMMON /MAIN1/ SN, VN
      COMMON/PLT/ SCALE, FL
      COMMON /DI/ CALCMP, TYPE
      INTEGER TYPE, CALCMP, FL, BND
      DATA CALCMP/6HCALCMP/
C
      READ 1, TYPE
    1 FORMAT(A6)
      IF(TYPE.EQ.CALCMP) GO TO 4
      READ 2, NTAPE, NCL
    2 FORMAT(2I5)
      READ 3, NPLT
    3 FORMAT(80I1)
C
      CALL RELOAD
      CALL IDI(FL)
      CALL DMAIN1(NTAPE,NCL,NPLT)
      GO TO 5
C
    4 CALL DMAIN2
    5 CALL EXIT
      END
```

```
VI FOR DMAIN1,DMAIN1
      SUBROUTINE DMAIN1(NTAPE,NCL,NPLT)
      DIMENSION BND(61,26), A(61,26), SN(61,26), VN(61,26), NPLT(80)
      DIMENSION XBP(100,4), YBP(100,4), BLN(4), B(100)
      DIMENSION XTEM(200), YTEM(200), FL(3000)
      DIMENSION MSG1(7), MSG2(5), MSG3(3), MSG4(2)
      DIMENSION IBF(100,4), JBF(100,4), NBPF(6)

      COMMON/MAIN/BND, A
      COMMON/MAIN1/ SN, VN
      COMMON/PLT/ SCALE, FL
      COMMON /D1/ CALCMP, TYPE
      INTEGER BND, FL, VAR, TYPE, CALCMP

      DATA MSG1/6HSOLUTI, 6HONS OF, 6H THE N, 6HAVIER , 6HSTOKES,
     1          6H EQUAT, 6HIONS  /
      DATA MSG2/6HSTREAM, 6H FUNCT, 6HION   /
      DATA MSG3/6HREYNOL, 6HDS NUM, 6HBER=  /
      DATA MSG4/6HFRAME , 6H NO.  /


      REWIND 4
C     NFRAME = 20
C     REWIND 2
C     WRITE(2) NFRAME
      DO 11 NR=1,NTAPE

      READ (4) M, N, NFILE, DT, TMAX, R, DR
      SCALE = 800./FLOAT(M-1)
      NWORD = N*M
      CALL INOUT(1,4,BND,NWORD)
      CALL INOUT(1,4,A,NWORD)
      READ(4) IBF, JBF, NBPF
      DO 1 I=1,NFILE
      READ (4) IK, (XBP(K,I), YBP(K,I), K=1,IK)
    1 BLN(I) = IK

      CALL IDT(FL)
      XX = FLOAT(M-1)
      DO 2 J=1,N
      YY = FLOAT(J-1)
      CALL PLOT1(0.0, YY, 3)
      CALL PLOT1(XX, YY, 2)
    2 CONTINUE
      YY = FLOAT(N-1)
      DO 3 I=1,M
      XX = FLOAT(I-1)
      CALL PLOT1(XX, 0.0, 3)
      CALL PLOT1(XX, YY, 2)
    3 CONTINUE
C
      CALL INTEN(2)

      DO 6 K1=1,NFILE
      K2 = BLN(K1)
      DO 4 I=1,K2
      XTEM(I) = XBP(I,K1)-1.
    4 YTEM(I) = YBP(I,K1)-1.
      CALL PLOT1(XTEM(1), YTEM(1), 3)
```

```
      DO 5 K3=1,K2
      CALL PLOT1(XTEM(K3), YTEM(K3), 2)
    5 CONTINUE
    6 CONTINUE
      CALL PLOT1(0.0, 0.0, 3)
      CALL INTEN(0)
      CALL SNDFLE(FL)
C     CALL LOCUSD(FL,IA)
C     FL(3000) = IA
C     CALL INOUT(0,2,FL,3000)

      K = 0
    7 K = K + 1
      IF(NPLT(K).EQ.2) GO TO 11

      READ(5) RR
      CALL INOUT(1,4,SN,NWORD)
      CALL INOUT(1,4,VM,NWORD)

      IF(NPLT(K).EQ.0) GO TO 7

      CALL IDI(FL)
      T1 = SN(1,1)
      T2 = T1
      DO 8 I=1,M
      DO 8 J=1,N
      IF(BND(I,J).GT.1) GO TO 8
      T1 = AMIN1(T1, SN(I,J))
      T2 = AMAX1(T2, SN(I,J))
    8 CONTINUE

C       GENERATE CONTOUR LEVELS

      RANGE = T2 - T1
      TS = (T2-T1)/FLOAT(NCL+1)
      B(1) = T1 + TS
      DO 9 K=2,NCL
    9 B(K) = B(K-1) + TS

      PRINT 10, T1, T2, (B(K), K=1,NCL)
   10 FORMAT(1H0, 10X, 16HMIN. GRID VALUE=,F15.4, 3X,
     1 11HMAX. VALUE= F15.4/ 10X, 14HCONTOUR LEVELS/(10X, 4F15.4))


      CALL INTEN(2)
      CALL LNTYPE(0)
      CALL LINE(FL,200,325,IA)
      CALL CHAR(FL,MSG1,IA)
      CALL LINE(FL,200,300,IA)
      CALL CHAR(FL,MSG2,IA)
      CALL LINE(FL,200,275,IA)
      CALL CHAR(FL,MSG3,IA)
      VAR = INBCD(IFIX(RR))
      CALL LINE(FL,465,275,IA)
      CALL CHAR(FL,VAR,IA)
      CALL LINE(FL,200,250,IA)
      CALL CHAR(FL,MSG4,IA)
      VAR = INBCD(K)
      CALL LINE(FL,395,250,IA)
```

```
      CALL CHAR(FL,VAR,IA)

C       PLOT SOLUTION SPACE BOUNDARY....

      CALL ENDPLT(XBP, YBP, FL, NFILS)

      CALL CONTOR(M,N,Z,XDIS,YDIS,NCL,SM,BND,A)
      CALL INTEG(0)
      CALL PLOT1(0.0, 0.0, 3)
      CALL SMOFLE(FL)
C     CALL LOCUSD(FL,IA)
C     CALL INOUT(0,2,FL,3000)
C     FL(3000) = IA
      GO TO 7

   11 CONTINUE
      CALL UNLOAD(4)
      RETURN
      END
```

```
*I FOR DMAIN2,DRAIN2
      SUBROUTINE DMAIN2
      DIMENSION BND(61,26), A(61,26), SN(61,26), VN(61,26), NPLT(80)
      DIMENSION XBP(100,4), YBP(100,4), BLN(4), FL(3000)
      DIMENSION IBF(100,4), JBF(100,4), NBPF(6)
C
      COMMON/OUT/IS, IVN, IU, IV, ITVP, IVEC, NCLS, NCL
      COMMON/MAIN/BND, A
      COMMON/MAIN1/ SN, VN
      COMMON/PLT/ SCALE, FL
C
      INTEGER BND, FL
C
C
      CALL IDPLOT
      CALL PLOT(2.0, 0.0, -3)
C
      READ 1, NCLS, NCL, IS, IVN, IU, IV, ITVP, IVEC
    1 FORMAT(8I5)
C
      READ 2, NTAPE
    2 FORMAT(I5)
C
      REWIND 4
      DO 6 NN=1,NTAPE
C
      READ 3, NPLT
    3 FORMAT(80I1)
      READ (4) M, N, NFILE, DT, TMAX, P, DR
      NWORD = N*M
      CALL INOUT(1,4,BND,NWORD)
      CALL INOUT(1,4,A,NWORD)
      READ(4) IBF, JBF, NBPF
      DO 4 I=1,NFILE
      READ (4) IK, (XBP(K,I), YBP(K,I), K=1,IK)
    4 BLN(I) = IK
C
      K = 0
    5 K = K + 1
      IF(NPLT(K).EQ.2) GO TO 6
C
      READ(4) RR
      CALL INOUT(1,4,SN,NWORD)
      CALL INOUT(1,4,VN,NWORD)
C
      IF(NPLT(K).EQ.0) GO TO 5
C
      XDIS = 5.0
      YDIS = XDIS*FLOAT(N-1)/FLOAT(M-1)
      SCALE = XDIS/FLOAT(M-1)
      CALL OUTPUT(M,N,SN,VN,A,DT,RR,XBP,YBP,BLN,NFILE,
     1            XDIS,YDIS,IBF,JBF,NBPF)
C
      GO TO 5
C
    6 CONTINUE
      CALL UNLOAD(4)
      CALL FIN1
      RETURN
```

```
C FOR CONTOUR,CONTOUR
      SUBROUTINE CONTOUR(M,N,XSCALE,XDIS,YDIS,NCL,G,BND,A)
      DIMENSION G(81,26), XX(5), YY(5), B(100), X(100), Y(100),
     1 BND(81,26), A(81,26), F(5), S(5), T(5)
      DIMENSION XP(200), YP(200)
      DIMENSION XS(6000), YS(6000)
      INTEGER BND
C
C        G= MATRIX OF GRID VALUES
C        M= NUMBER OF COLUMNS IN G
C        N= NUMBER OF ROWS IN G
C        XSCALE= SIZE OF PLOT (INCHES ALONG X-AXIS)
C        C= ARRAY OF CONTOUR LEVELS
C        NCL= NUMBER OF LEVELS IN B
C        BND= BOUNDARY ARRAY, 0 = INTERIOR POINT, 1 = BOUNDARY POINT,
C        2 = EXTERIOR POINT
C
C
      DO 1 I=1,100
      X(I) = I-1
    1 Y(I) = I-1
C
C
      M1 = M - 1
      N1 = N - 1
C
      LS = 0
      DO 44 I=1,M1
C
      NS = 1
C
    2 JTEMP = 1
      DO 5 J=NS,N
      GO TO (3,4), JTEMP
    3 IF(BND(I,J+1).GE.1 .AND. BND(I+1,J+1).GE.1) GO TO 5
      NB = J
      JTEMP = 2
      GO TO 5
    4 IF(BND(I,J+1).EQ.0 .OR. BND(I+1,J+1).EQ.0) GO TO 5
      NF = J
      NS = J + 1
      IF(NS.GT.N) NS = N
      GO TO 6
    5 CONTINUE
      GO TO 44
C
C
    6 DO 43 J=NB,NF
C
      ICL = 1
      IS = 0
      IF(BND(I,J).GE.1) IS = IS + 8
      IF(BND(I,J+1).GE.1) IS = IS + 2
      IF(BND(I+1,J+1).GE.1) IS = IS + 3
      IF(BND(I+1,J).GE.1) IS = IS + 4
      IF(A(I+1,J)-1.) 11, 7, 11
    7 IF(A(I,J)-1.) 11,8,11
    8 IF(A(I+1,J+1)-1.) 11,9,11
    9 IF(A(I+1,J+1)-1.) 11,10,11
```

```
   10 IF(BND(I,J).EQ.2.OR.BND(I,J+1).EQ.2.OR.BND(I+1,J+1).EQ. 2
     1    .OR.BND(I+1,J).EQ.2) GO TO 11                                    261
      IS = 1
      GO TO 13
C
   11 IF(IS.GE.1 .AND. IS.LE.15) GO TO 13
      IS = 1
      PRINT 12, I, J
   12 FORMAT(1H , 20X, 31HWARNING... 'IS' IS OUT OF RANGE, 3X, 2HI=I3,
     1 3X, 2HJ=I3)
C
   13 F(1) = G(I,J)
      F(2) = G(I,J+1)
      F(3) = G(I+1,J+1)
      F(4) = G(I+1,J)
      S(1) = X(I)
      S(2) = X(I)
      S(3) = X(I+1)
      S(4) = X(I+1)
      T(1) = Y(J)
      T(2) = Y(J+1)
      T(3) = Y(J+1)
      T(4) = Y(J)
      IN = 4
C
      GO TO (28,14,15,16,17,18,19,20,26,21,27,24,23,43,43), IS
C
   14 F(2) = G(I,J+1)
      F(3) = G(I,J+1)
      F(4) = G(I+1,J+1)
      F(5) = G(I+1,J)
      S(3) = X(I+1) - A(I,J+1)
      S(5) = X(I+1)
      T(2) = Y(J) + A(I,J+1)
      T(4) = Y(J+1)
      T(5) = Y(J)
      IN = 5
      GO TO 28
C
   15 F(3) = G(I+1,J+1)
      F(4) = G(I+1,J+1)
      F(5) = G(I+1,J)
      S(3) = X(I) + A(I+1,J+1)
      S(5) = X(I+1)
      T(4) = Y(J) + A(I+1,J+1)
      T(5) = Y(J)
      IN = 5
      GO TO 28
C
   16 F(4) = G(I+1,J)
      F(5) = G(I+1,J)
      S(5) = X(I) + A(I+1,J)
      T(4) = Y(J+1) - A(I+1,J)
      T(5) = Y(J)
      IN = 5
      GO TO 28
C
   17 F(2) = G(I,J+1)
      F(3) = G(I+1,J+1)
```

```
      T(2) = Y(J) + A(I,J+1)
      T(3) = Y(J) + A(I+1,J+1)
      IN = 4
      GO TO 28
C
   18 IF(BND(I,J+1).GE.1 .AND. BND(I+1,J).GE.1) GO TO 25
      F(1) = G(I,J)
      F(5) = G(I,J)
      S(5) = X(I+1) - A(I,J)
      T(1) = Y(J+1) - A(I,J)
      T(5) = Y(J)
      IN = 5
      GO TO 28
C
   19 F(3) = G(I+1,J+1)
      F(4) = G(I+1,J)
      S(3) = X(I) + A(I+1,J+1)
      S(4) = X(I) + A(I+1,J)
      IN = 4
      GO TO 28
C
   20 F(1) = G(I,J)
      F(2) = G(I,J+1)
      S(1) = X(I+1) - A(I,J)
      S(2) = X(I+1) - A(I,J+1)
      IN = 4
      GO TO 28
C
   21 F(1) = G(I,J)
      F(4) = G(I+1,J)
      T(1) = Y(J+1) - A(I,J)
      T(4) = Y(J+1) - A(I+1,J)
      IN = 4
      GO TO 28
C
   22 ICL = 4
   23 F(1) = G(I,J)
      F(3) = A(I+1,J+1)*(G(I+1,J+1) - G(I,J+1)) + G(I,J+1)
      S(3) = X(I) + A(I+1,J+1)
      T(1) = Y(J+1) - A(I,J)
      IN = 3
      GO TO 28
C
   24 F(1) = G(I,J+1)
      F(2) = G(I+1,J+1)
      F(3) = G(I+1,J)
      S(1) = X(I+1) - A(I,J+1)
      S(2) = X(I+1)
      T(1) = Y(J+1)
      T(3) = Y(J+1) - A(I+1,J)
      IN = 3
      ICL = 1
      GO TO 28
C
   25 ICL = 2
   26 IF(BND(I,J).GE.1 .AND. BND(I+1,J+1).GE.1) GO TO 22
      F(2) = G(I,J+1)
      F(3) = G(I+1,J)
      S(3) = X(I) + A(I+1,J)
```

```
      T(2) = Y(J) + A(I,J+1)
      T(3) = Y(J)
      IN = 3
      GO TO 29
C
   27 F(1) = G(I,J)
      F(2) = G(I+1,J+1)
      F(3) = G(I+1,J)
      S(1) = X(I+1) - A(I,J)
      S(2) = X(I+1)
      T(2) = Y(J) + A(I+1,J+1)
      T(3) = Y(J)
      IN = 3
      ICL = 1
C
C
   28 DO 42 L=1,NCL
      LL = 0
      ISW = 1
C
      I1 = 1
      I2 = 2
   29 G1 = F(I1)
      G2 = F(I2)
      IF(B(L)-G1) 30, 37, 31
   30 IF(B(L)-G2) 34, 37, 32
   31 IF(G2-B(L)) 34, 37, 32
   32 LL = LL + 1
      IF(ABS(S(I2)-S(I1)) .LT. 0.0001) GO TO 33
      SLOPE = (T(I2)-T(I1))/(S(I2)-S(I1))
      IF(ABS(SLOPE) .GT. 1.0) GO TO 33
C      TOP AND BOTTOM....
      XX(LL) = (S(I2)-S(I1))*(B(L)-G1)/(G2-G1) + S(I1)
      YY(LL) = (XX(LL)-S(I1))*SLOPE + T(I1)
      GO TO 34
C      FRONT AND OPPOSITE....
   33 SLOPE = (S(I2)-S(I1))/(T(I2)-T(I1))
      YY(LL) = (T(I2)-T(I1))*(B(L)-G1)/(G2-G1) + T(I1)
      XX(LL) = (YY(LL)-T(I1))*SLOPE + S(I1)
   34 I1 = I1 + 1
      I2 = I2 + 1
      GO TO (35, 36), ISW
   35 IF(I2.LE.IN) GO TO 29
      I1 = 1
      I2 = IN
      ISW = 2
      GO TO 29
C
C
   36 IF(LL.EQ.0) GO TO 42
      IF(LL.EQ.2) GO TO 39
   37 PRINT 38, LL, B(L), I, J, (F(I1), I1=1,4)
   38 FORMAT(1H0, 20X, 14HWARNING... LL=I2, 3X, 2HB=F12.6, 3X,
     1  2HI=I2, 3X, 2HJ=I2, 20X, 7HA(I,J)=, F12.6, 3X, 9HA(I,J+1)=,
     2  F12.6, 3X, 11HA(I+1,J+1)=F12.6, 3X, 9HA(I+1,J)=F12.6)
      GO TO 42
C
   39 IF(XX(1).LT.XX(2)) GO TO 40
      TEM = XX(1)
```

```
      XX(1) = XX(2)
      XX(2) = TEM
      TEM = YY(1)
      YY(1) = YY(2)
      YY(2) = TEM
   40 LS = LS + 1
      XS(LS) = XX(1)
      YS(LS) = YY(1)
      LS = LS + 1
      XS(LS) = XX(2)
      YS(LS) = YY(2)
      IF(LS.LT.6000) GO TO 42
      PRINT 41
   41 FORMAT(1H0, 30X, 42HDIMENSION ON XS IS TOO SMALL IN SUB CONTOR)
      RETURN
C
   42 CONTINUE
      GO TO (43,24,43,27), ICL
   43 CONTINUE
      GO TO 2
   44 CONTINUE
C
C
C
C     PLOT CONTOR LINES IN LINE MODE RATHER THAN COLUMN MODE.....
C
      ISW = -1
      EPS = .001
   45 L = 1
      IPS = 1
      JJ = 0
C
      JJ = JJ + 1
      XP(JJ) = XS(L)
      YP(JJ) = YS(L)
   46 JJ = JJ + 1
      IF(JJ.LT.200) GO TO 48
      PRINT 47
   47 FORMAT(1H ,20X, 46HDIMENSION ON XP,YP HAS BEEN EXCEEDED IN CONTOR)
      CALL EXIT
C
   48 XP(JJ) = XS(L+1)
      YP(JJ) = YS(L+1)
C
      K = J
   49 K = K + 1
      IF(K.EQ.L+1) GO TO 55
      IF(ABS(XS(K)-XS(L+1)).GT.EPS .OR.
     1   ABS(YS(K)-YS(L+1)).GT.EPS) GO TO 55
C
C     DELETE XS(L), YS(L), AND XS(L+1), YS(L+1) FROM STACK....
C
   50 LS = LS - 2
      IF(LS.LE.0) GO TO 56
      L1 = L
C
      L = K - 2
      K1 = K
      IF((-1.)**K.LT.0.0) GO TO 51
```

```
      TEM = XS(K-1)
      XS(K-1) = XS(K)
      XS(K) = TEM
      TEM = YS(K-1)
      YS(K-1) = YS(K)
      YS(K) = TEM
      L = K - 3
      K1 = K - 1
   51 IF(K.GT.L1+1) GO TO 52
      L = K1
C
   52 IF(L.GT.0) GO TO 53
      L = 1
   53 DO 54 LL=L1,LS
      XS(LL) = XS(LL+2)
   54 YS(LL) = YS(LL+2)
C
      GO TO (45,56), IPS
C
   55 IF(K.LT.LS .AND. X(K).LT.FLOAT(M)) GO TO 49
      IPS = 2
      L1 = L
      LS = LS - 2
      GO TO 53
C
C      PLOT CONTOR LINE....
C
   56 J1 = (JJ-1)*(1+ISW)/2 + 1
      CALL LNPLT(XP,YP,JJ)
      CALL PLOT1(XP(J1), YP(J1), 3)
      DO 59 J=2,JJ
      IF(ISW.LT.0) GO TO 57
      J1 = JJ + 1 - J
      GO TO 58
   57 J1 = J
   58 CALL PLOT1(XP(J1), YP(J1), 2)
   59 CONTINUE
      IF(LS.LE.0) RETURN
      ISW = -ISW
      GO TO 45
C
C
      END
```

```
C1 FOR VELVEC,VELVEC
      SUBROUTINE VELVEC(M,N,SN,AND)
C
      DIMENSION SN(61,26), AND(61,26)
C
      INTEGER AND
      REAL MAXMAG
C
      MAXMAG = 0.0
      DX = 1./FLOAT(N-1)
      M2 = 1./FLOAT(N-1)
      N1 = N-1
      M1 = M-1
      M2 = 3 - 2
      DY = DX
      SCALE = 7.0
C
      PHI = 0.1E0
      COSPI = COS(PHI)
      SINPI = SIN(PHI)
      CALL PLOT(5.0, 2.0, -3)
C
      K = 0
      IF((-1)**M.LT.0) GO TO 1
      IC = 0
      GO TO 2
    1 IC = 1
    2 DO 8 I=2,M2,2
      K = K + 1
      IF((-1)**K.GT.0) GO TO 3
      J1 = 2 + IC
      ISW = 1
      GO TO 4
    3 J1 = 3 - IC
      ISW = 2
    4 DO 8 L=J1,N1,2
      GO TO (5,6), ISW
    5 J = L
      GO TO 7
    6 J = N1 + J1 - L
    7 IF(SN(I,J).NE.0) GO TO 8
      U = .5/M2*(SN(I,J+1) - SN(I,J-1))
      V = -.5/M2*(SN(I+1,J) - SN(I-1,J))
      TV = SQRT(U*U + V*V)
      MAXMAG = AMAX1(MAXMAG, TV)
    8 CONTINUE
      PRINT 9, MAXMAG
    9 FORMAT(1H // 30X, 7HMAXMAG=, F10.3)
      VSC = 3.0*DX*SCALE/MAXMAG
C
      DO 16 I=2,M2,2
      K = K + 1
      IF((-1)**K.GT.0) GO TO 10
      J1 = 2 + IC
      ISW = 1
      GO TO 11
   10 J1 = 3 - IC
      ISW = 2
C
```

```
   11 DO 15 L=J1,M1,2
C
      GO TO (12,13), ISW
   12 J = L
      GO TO 14
   13 J = M1 + J1 - L
C
   14 IF(BNE(I,J).NE.0) GO TO 15
      X1 = SCALE*(FLOAT(I-1)*DX)
      Y1 = SCALE*(FLOAT(J)*DY)
      CALL PLOT(-Y1, X1, 3)
      U = .5/H2*(SN(I,J+1) - SN(I,J-1))
      V = -.5/H2*(SN(I+1,J) - SN(I-1,J))
       X2 = X1 + VSC*U
      Y2 = Y1 + VSC*V
      CALL PLOT(-Y2, X2, 2)
      X3 = X1 + .65*VSC*(COSPI*U - SINPI*V)
      Y3 = Y1 + .65*VSC*(COSPI*V + SINPI*U)
      CALL PLOT(-Y3, X3, 2)
      X4 = X1 + .75*VSC*U
      Y4 = Y1 + .75*VSC*V
      CALL PLOT(-Y4, X4, 2)
C
   15 CONTINUE
   16 CONTINUE
      RETURN
      END
```

AD-A038 694 UTAH UNIV SALT LAKE CITY DEPT OF COMPUTER SCIENCE F/G 6/5
AN APPLICATION OF COMPUTER GRAPHICS: TWO CONCURRENT INVESTIGATI--ETC(U)
NOV 71 H GREENFIELD, R DEBRY F30602-70-C-0300
UNCLASSIFIED UTEC-CSC-71-115 NL

4 OF 4
AD A038694

END
DATE FILMED
5-77

```
.I FOR PLOT1,PLOT1
      SUBROUTINE PLOT1(A,B,KK)
      COMMON/PLT/ SCALE, FL
      COMMON/PLT1/ ISW, IX1, IY1
      COMMON /D1/ CALCMP, TYPE
      INTEGER TYPE, CALCMP
      INTEGER FL(3000)
      IF(TYPE.NE.CALCMP) GO TO 1

      X = A*SCALE
      Y = B*SCALE
      CALL PLOT(X,Y,KK)
      GO TO 5

    1 IF(KK.LT.0) GO TO 5
      IX = SCALE*A + 100.
      IY = SCALE*B + 400.
      GO TO (2,2,4), KK
    2 IF(KK.EQ.ISW) GO TO 3
      CALL LNTYPE(0)
      CALL LINE(FL,IX1,IY1,IA)
      CALL LNTYPE(3)
      ISW = 2
    3 CALL LINE(FL,IX,IY,IA)
      GO TO 5
    4 ISW = 3
      IX1 = IX
      IY1 = IY

    5 RETURN
      END
```

```
*1 FOR LNPLT,LNPLT
      SUBROUTINE LNPLT(X,Y,KPTS)                                    269
      DIMENSION X(200), Y(200)
C
      EPS = .025
      K = 0
C
    1 K = K + 1
      IF(K+2.GE.KPTS) RETURN
      K1 = K + 1
      K2 = K + 2
    2 A = Y(K) - Y(K2)
      B = X(K2) - X(K)
      C = X(K)*Y(K2) - X(K2)*Y(K)
      DEM = SQRT(A*A + B*B)
      DIS = ABS((A*X(K1) + B*Y(K1) + C)/DEM)
      IF(DIS.GT.EPS) GO TO 1
C
C       ELIMINATE POINT FROM LINE
C
      KPTS = KPTS - 1
      DO 3 I=K1,KPTS
      X(I) = X(I+1)
    3 Y(I) = Y(I+1)
    4 IF(K1.LT.KPTS) GO TO 1
      RETURN
C
      END
```

```
      END
$I FOR BNDPLT,BNDPLT
      SUBROUTINE BNDPLT(XBP, YBP, BLN, NFILE)
      DIMENSION XBP(100,4), YBP(100,4), BLN(4), XTEM(100), YTEM(100)
      COMMON/COMS/ XTEM, YTEM
C
C       PLOT BOUNDARY CONFIGURATION FROM OPTICAL DISPLAY INPUT....
C
      DO 3 K1=1,NFILE
      K2 = BLN(K1)
      DO 1 I=1,K2
      XTEM(I) = XBP(I,K1) -1.
    1 YTEM(I) = YBP(I,K1) -1.
      CALL PLOT1(XTEM(1), YTEM(1), 3)
      CALL UNPLT(XTEM,YTEM,K2)
      DO 2 K3=1,K2
      CALL PLOT1(XTEM(K3), YTEM(K3), 2)
    2 CONTINUE
    3 CONTINUE
C
      RETURN
      END
```

```
*I FOR OUTPUT,OUTPUT
      SUBROUTINE OUTPUT(M,N,SN,VN,NFRAME,DT,RR,XBP,YBP,BLN,NFILE,      271
     1                  XDIS,YDIS,IBF,JBF,NBPF)
         DIMENSION IBF(100,4), JBF(100,4), NBPF(6)
      DIMENSION  B(100), TA(61,26), SN(61,26), BND(61,26),
     1 VN(61,26), A(61,26), FL(3000)
      DIMENSION XBP(100,4), YBP(100,4), BLN(4), Y1(100,4)

      COMMON/MAIN/BND, A
      COMMON/OUT/IS, IVN, IO, IV, ITVP, IVEC, NCLS, NCL
      COMMON/PLT/ SCALE, FL

      INTEGER SNPLOT, VNPLOT, UPLOT, VPLOT, TVPLOT, BND, FL


      M1 = M-1
      N1 = N-1
      DX = 1./FLOAT(N1)
    1 SNPLOT = IS
      VNPLOT = IVN
      UPLOT = IO
      VPLOT = IV
      TVPLOT =ITVP
      IVECT = IVEC
      IF(SNPLOT.EQ.0) GO TO 4
    2 K = K + 1
      DO 3 I=1,M
      DO 3 J=1,N
    3 TA(I,J) = SN(I,J)
      CALL PLOT(0.0, 0.0, -3)
      CALL SYMBL4(3.0, 2.0, 0.1, 15HSTREAM FUNCTION, 0.0, 15)
      ISTR = 1
      GO TO 24

    4 IF(VNPLOT.EQ.0) GO TO 7
    5 VNPLOT = 0
      DO 6 I=1,M
      DO 6 J=1,N
    6 TA(I,J) = VN(I,J)
      CALL PLOT(0.0, 0.0, -3)
      CALL SYMBL4(3.0,2.0,0.1,18HVORTICITY FUNCTION, 0.0, 18)
      GO TO 24

    7 IF(UPLOT.EQ.0) GO TO 11
    8 UPLOT = 0

      DO 10 I=1,M
      DO 10 J=1,N
      IF(BND(I,J).EQ.0) GO TO 9
      TA(I,J) = 0.0
      GO TO 10
    9 TA(I,J) =  .5/DX*(SN(I,J+1) - SN(I,J-1))
   10 CONTINUE
      CALL BNDARY(M,N,M1,N1,IBF,JBF,NBPF,TA)
      CALL PLOT(0.0, 0.0, -3)
      CALL SYMBL4(3.0,2.0,0.1,10HU VELOCITY, 0.0, 10)
      GO TO 24

   11 IF(VPLOT.EQ.0) GO TO 16
```

```
   12 VPLOT = 0

   13 DO 15 I=1,M
      DO 15 J=1,N
      IF(BND(I,J).EQ.0) GO TO 14
      TA(I,J) = 0.0
      GO TO 15
   14 TA(I,J) = -0.5/DX*(SN(I+1,J)-SN(I-1,J))
   15 CONTINUE
      CALL BNDARY(M,N,M1,N1,IBF,JBF,NBPF,TA)

      CALL PLOT(0.0, 0.0, -3)
      CALL SYMBL4(3.0,2.0,0.1,10HV VELOCITY, 0.0, 10)
      GO TO 24

   16 IF(IVPLOT.EQ.0) GO TO 20
   17 IVPLOT = 0

      DO 19 I=1,M
      DO 19 J=1,N
      IF(BND(I,J).EQ.0) GO TO 18
      GO TO 19
   18 T1 = -.5/DX*(SN(I,J+1) - SN(I,J-1))
      T2 =  .5/DX*(SN(I+1,J) - SN(I-1,J))
      TA(I,J) = SQRT(T1*T1 + T2*T2)
   19 CONTINUE
      CALL BNDARY(M,N,M1,N1,IBF,JBF,NBPF,TA)

      CALL PLOT(0.0, 0.0, -3)
      CALL SYMBL4(3.0,2.0,0.1,14HTOTAL VELOCITY, 0.0, 14)
      GO TO 24

   20 IF(IVECT.EQ.0) GO TO 32
   21 IVECT = 0
      CALL SYMBL4(6.0, 4.7, 0.1, 15HVELOCITY VECTOR, 90.0, 15)
      CALL VELVEC(M,N,SN,BND)
      TEM = SCALE
      SCALE = 7.0/FLOAT(M-1)
      DO 23 K=1,NFILE
      L = BLN(K)
      DO 22 KK=1,L
   22 Y1(KK,K) = -YBP(KK,K) + 1.0
   23 CONTINUE
      CALL BNDPLT(Y1, XBP, BLN, NFILE)
      SCALE = TEM
      CALL PLOT(9.0, -2.75, -3)
      RETURN


   24 CALL SYMBL4(3.0, 1.7, .1, 2HR=,0.0,2)
      CALL NUMBER(3.40, 1.7, .1, RR, 0.0, 0)
      CALL SYMBL4(3.0, 1.4, .1, 10HFRAME NO. ,0.0,10)
      FRAME = NFRAME
      CALL NUMBER(4.2, 1.4, .1, FRAME, 0.0, 0)

C      CALCULATE MAXIMUM AND MINIMUM OF GRID VALUES
   25 T1 = TA(1,1)
      T2 = T1
      DO 26 I=1,M
```

```
      DO 26 J=1,N
      IF(BND(I,J).GT.1) GO TO 26
      T1 = AMIN1(T1, TA(I,J))
      T2 = AMAX1(T2, TA(I,J))
   26 CONTINUE

C      GENERATE CONTOUR LEVELS

      NCL1 = NCL
      IF(ISTR.EQ.1) NCL1 = NCLS
      TS = (T2-T1)/FLOAT(NCL1+1)
      B(1) = T1 + TS
      DO 27 K=2,NCL1
   27 B(K) = B(K-1) + TS

   28 PRINT 29, T1, T2, (B(K), K=1,NCL1)
   29 FORMAT(1H0, 10X, 16HMIN. GRID VALUE=,F15.4, 3X,
     1 11HMAX. VALUE= F15.4/ 10X, 14HCONTOUR LEVELS/(10X, 4F15.4))


   30 CALL PLOT(1.0, 5.3, -3)
      CALL ISOPLT(M,N,XDIS,YDIS,TA,BND,A)
      CALL PLOT(.4, 2.2, -3)

   31 CALL CONTOR(M, N, B, XDIS, YDIS, NCL1, TA, BND, A)
      CALL BNDPLT(XBP, YBP, BLN, NFILE)

      ISTR = 0
      CALL PLOT(9.0, -7.5, -3)
      IF(VNPLOT.EQ.1) GO TO 5
      IF(UPLOT.EQ.1) GO TO 8
      IF(VPLOT.EQ.1) GO TO 12
      IF(TVPLOT.EQ.1) GO TO 17
      IF(IVECT.EQ.1) GO TO 21
   32 RETURN


      SUBROUTINE BNDARY(M,N,M1,N1,IBF,JBF,NBPF,TA)
      DIMENSION IBF(100,4), JBF(100,4), NBPF(6), TA(61,26)
      DO 37 KK=1,4
      L = NBPF(KK)
      IF(L.EQ.0) GO TO 37
      DO 36 K=1,L
      I = IBF(K,KK)
      J = JBF(K,KK)
      IF(J.NE.1) GO TO 33
      TA(I,1) = TA(I,2)
      GO TO 36
   33 IF(J.NE.N) GO TO 34
      TA(I,N) = TA(I,N1)
      GO TO 36
   34 IF(I.NE.1) GO TO 35
      TA(1,J) = TA(2,J)
      GO TO 36
   35 IF(I.NE.M) GO TO 36
      TA(M,J) = TA(M1,J)
   36 CONTINUE
   37 CONTINUE
      RETURN
```

```
VI FOR ISOPLT,ISOPLT
      SUBROUTINE ISOPLT(M,N,XP,YP,TA,BND,A)
      DIMENSION BND(61,26), A(61,26)
      DIMENSION TA(61,26), U(75), V(75), T(75), S(75), TEM(75)
      DIMENSION UA(75), VA(75)
      REAL M1, M2
      LOGICAL DOWN, FIRST
      DIMENSION FL(3000)
      COMMON/PLT/ SCALE, FL
      COMMON /D1/ CALCMP, TYPE
      INTEGER BND, FL, TYPE, CALCMP

      DO 1 I=1,M
      DO 1 J=1,N
    1 TA(I,J) = ABS(TA(I,J))
      M1 = M + 1
      M2 = M + 2
      DY = .80*YP/FLOAT(N-1)
      DX = DY
      X = XP + DX
      ANG = 0.86602540
      ER = 0.2*ANG*DX
      ZSCALE = 0.0
      DO 2 J=1,N
      DO 2 I=1,M
      IF(BND(I,J).EQ.2) GO TO 2
      ZSCALE = AMAX1(ZSCALE, ABS(TA(I,J)))
    2 CONTINUE

      DO 30 II=1,M
      I = M + 1 - II

      X = X - DX
      Y = -DY

      U(1) = ANG*X
      V(1) = -.5*X
      UA(1) = 0.0
      VA(1) = 0.0
      JJ = 1

      NN = N - 1
      DO 6 J=1,NN
      JJ = JJ + 1
      UA(JJ) = 0.0
      VA(JJ) = 0.0
      Y = Y + DY
      U(JJ) = ANG*(X + Y)
      IF(BND(I,J).EQ.0) GO TO 3
      IF(BND(I,J).EQ.1) GO TO 5
      V(JJ) = .5*(Y-X)
      IF(J.EQ.N) GO TO 6
      IF(BND(I,J+1).EQ.2) GO TO 6
      UA(JJ) = ANG*DY*(1.-A(I,J+1))
      VA(JJ) = .5*DY*(1.-A(I,J+1))
      JJ = JJ + 1
      U(JJ) = U(JJ-1)
      V(JJ) = .5*(Y-X) + TA(I,J+1)/ZSCALE
      UA(JJ) = UA(JJ-1)
```

```
      VA(JJ) = VA(JJ-1)
      GO TO 6
    3 V(JJ) = .5*(Y-X) + TA(I,J+1)/ZSCALE
      UA(JJ) = ANG*DY*(1.-A(I,J+1))
      VA(JJ) = .5*DY*(1.-A(I,J+1))
      IF(J.EQ.NN) GO TO 4
      IF(BND(I,J+1).EQ.0) GO TO 6
      IF(J+2.GT.N) GO TO 6
      IF(BND(I,J+2).LT.2) GO TO 6
    4 JJ = JJ + 1
      U(JJ) = U(JJ-1)
      V(JJ) = .5*(Y-X)
      UA(JJ) = UA(JJ-1)
      VA(JJ) = VA(JJ-1)
      GO TO 6
    5 IF(BND(I,J+1).EQ.2) TA(I,J+1) = 0.0
      V(JJ) = .5*(Y-X) + TA(I,J+1)/ZSCALE
      UA(JJ) = ANG*DY*(1.-A(I,J+1))
      VA(JJ) = .5*DY*(1.-A(I,J+1))
      IF(J.LT.NN) GO TO 6
      JJ = JJ + 1
      U(JJ) = U(JJ-1)
      V(JJ) = .5*(Y-X)
    6 CONTINUE
      DOWN = .TRUE.

      IF(II.GT.1) GO TO 9
C      PLOT FIRST LINE WITHOUT HIDDEN LINE ALGORITHM...
      CALL PLOT1(U(1), V(1), 3)
      DO 7 J=2,JJ
    7 CALL PLOT1(U(J)-UA(J), V(J)-VA(J), 2)
      DO 8 K=2,JJ
    8 T(K) = -10.0
      GO TO 22

    9 CALL PLOT1(U(1), V(1), 3)
      DO 10 K=2,3
   10 CALL PLOT1(U(K)-UA(K), V(K)-VA(K), 2)
      S(1) = U(1)
      T(1) = V(1)
      FIRST = .FALSE.
      IF(V(4).LT.T(3)) FIRST = .TRUE.

      DO 21 K=4,JJ
      DO 12 KK=1,N1
      IF(ABS(S(KK)-U(K)).GT.ER) GO TO 12
      IF(V(K)-T(KK)) 11, 14, 14
   11 IF(DOWN) GO TO 15
      CALL PLOT1(U(K), V(K), 3)
      DOWN = .FALSE.
      GO TO 21
   12 CONTINUE
      PRINT 13, K
   13 FORMAT(1H0, 30X, 8HHELP....., 2HL=I3)
      RETURN

   14 FIRST = .FALSE.
   15 IF(.NOT. DOWN) GO TO 16
      CALL PLOT1(U(K)-UA(K), V(K)-VA(K), 2)
```

```
      GO TO 21

   16 M1 = (T(KK) - T(KK-1))/(S(KK) - S(KK-1))
      IF(ABS(U(K)-U(K-1)).GT.ER) GO TO 17
      SS = U(K)
      GO TO 18
   17 M2 = (V(K) - V(K-1))/(U(K) - U(K-1))
      SS = (M2*U(K-1) - M1*S(KK-1) + T(KK-1) - V(K-1))/(M2 - M1)
   18 TT = M1*(SS - S(KK-1)) + T(KK-1)
      IF(S(KK-1)-ER.GT.SS .OR. SS.GT.S(KK)+ER) GO TO 20
      IF(DOWN) GO TO 19
      CALL PLOT1(SS, TT, 3)
      CALL PLOT1(U(K)-UA(K), V(K)-VA(K), 2)
      DOWN = .TRUE.
      GO TO 21
   19 CALL PLOT1(SS-UA(K), TT, 2)
   20 CALL PLOT1(U(K)-UA(K), V(K)-VA(K), 3)
      DOWN = .FALSE.

   21 CONTINUE

   22 S(2) = U(1)
      DO 24 K=2,N
      DO 23 J=K,JJ
      IF(U(J).GT.S(K)+ER) GO TO 24
   23 CONTINUE
   24 S(K+1) = U(J)
C        STORE THE MAXIMUM VALUE OF THE V(K) LINE IN ARRAY T(K)
      T(1) = V(1)
      DO 27 K=2,N1
      TEM(K) = -10.0
      DO 26 KK=K,JJ
      IF(U(KK)-S(K)) 26, 25, 26
   25 TEM(K) = AMAX1(V(KK), T(K-1), TEM(K))
   26 CONTINUE
   27 CONTINUE
      DO 28 K=2,N1
   28 T(K) = TEM(K)
   29 CONTINUE
   30 CONTINUE

      IF(TYPE.NE.CALCMP) CALL SNDFLE(FL)

      RETURN

      SUBROUTINE PLOT1(A,B,KK)
      DIMENSION FL(3000)
      COMMON /D1/ CALCMP, TYPE
      COMMON/PLT/ SCALE, FL
      COMMON/PLT2/ ISW, IX1, IY1
      INTEGER TYPE, CALCMP, FL

      IF(TYPE.NE.CALCMP) GO TO 31

      CALL PLOT(A,B,KK)
      GO TO 35

   31 IF(KK.LT.0) GO TO 35
      IX = 70.0*A + 300.0
```

```
      IY = 70.0*B + 600.0
      GO TO (32,32,34), KK
   32 IF(KK.EQ.ISW) GO TO 33
      CALL LNTYPE(0)
      CALL LINE(FL,IX1,IY1,IA)
      CALL LNTYPE(3)
      ISW = 2
   33 CALL LINE(FL,IX,IY,IA)
      GO TO 35
   34 ISW = 3
      IX1 = IX
      IY1 = IY

   35 RETURN
      END
```