

CACHE 2010 Award Lecture:  
Computing Applications in Ch.E. Education

# Computing in Chemical Engineering:

From Mainframes to Main Street

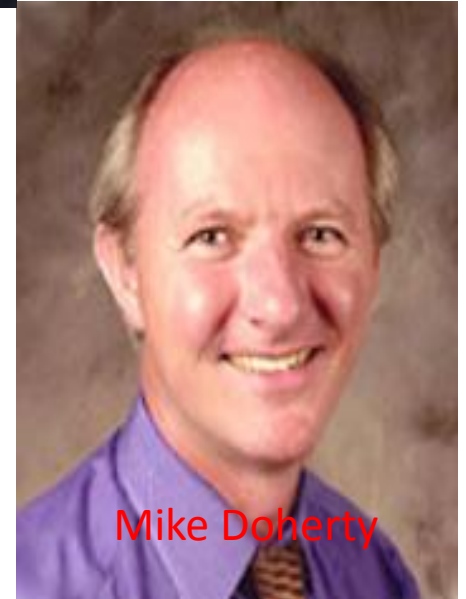
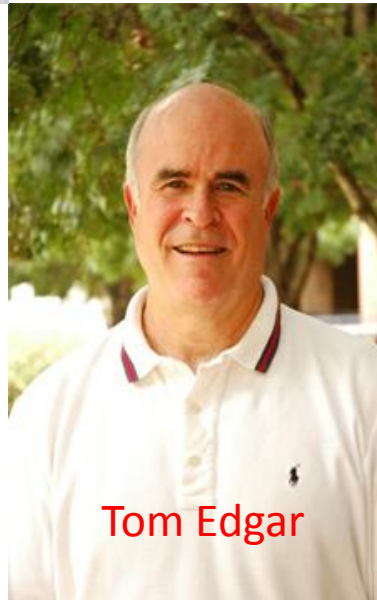
Duncan A. Mellichamp

Professor Emeritus

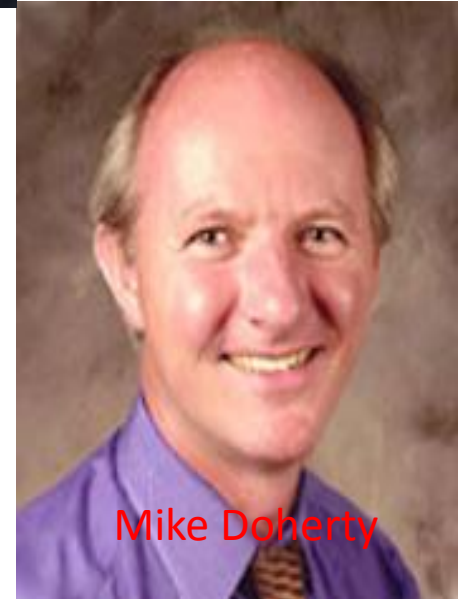
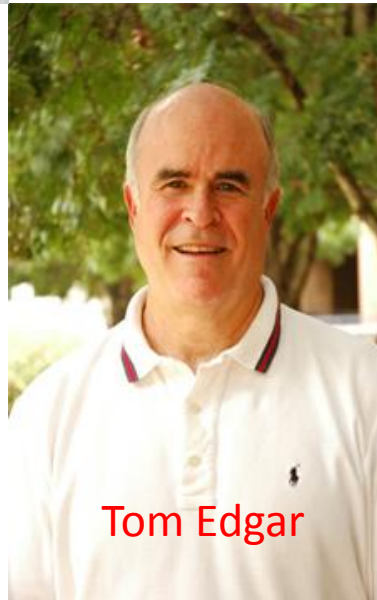
Department of Chemical Engineering

University of California, Santa Barbara

Colleagues to Whom I Owe a Lot ... & *Three Out of Four* Are Nice Guys!



**OK, I Lied ... All Four Are Great. Everyone Should Have Such Colleagues!!**



## Career Timeline

- '54-'59 Georgia Tech; B.S. Chemical Engineering (Co-operative Plan)
- '59-'60 Technische Hochschule Stuttgart (Germany); Exchange Fellowship
- '60-'64 Purdue University; Ph.D. Chemical Engineering
- '64-'66 DuPont Textile Fibers; Kinston, NC; Research Engineer
- '66-'03 UC Santa Barbara; Dep't Chemical Engineering, Founding Faculty
- '03- Retired ... but didn't quit!
  - \* Individual research (Profitability Measures)
  - \* Co-teaching design course (*pro bono*) with Mike Doherty
  - \* Board service, non-profits: UCSB Foundation
    - Opera Santa Barbara (presently, president of the board)

Q: Did it all really happen this way?

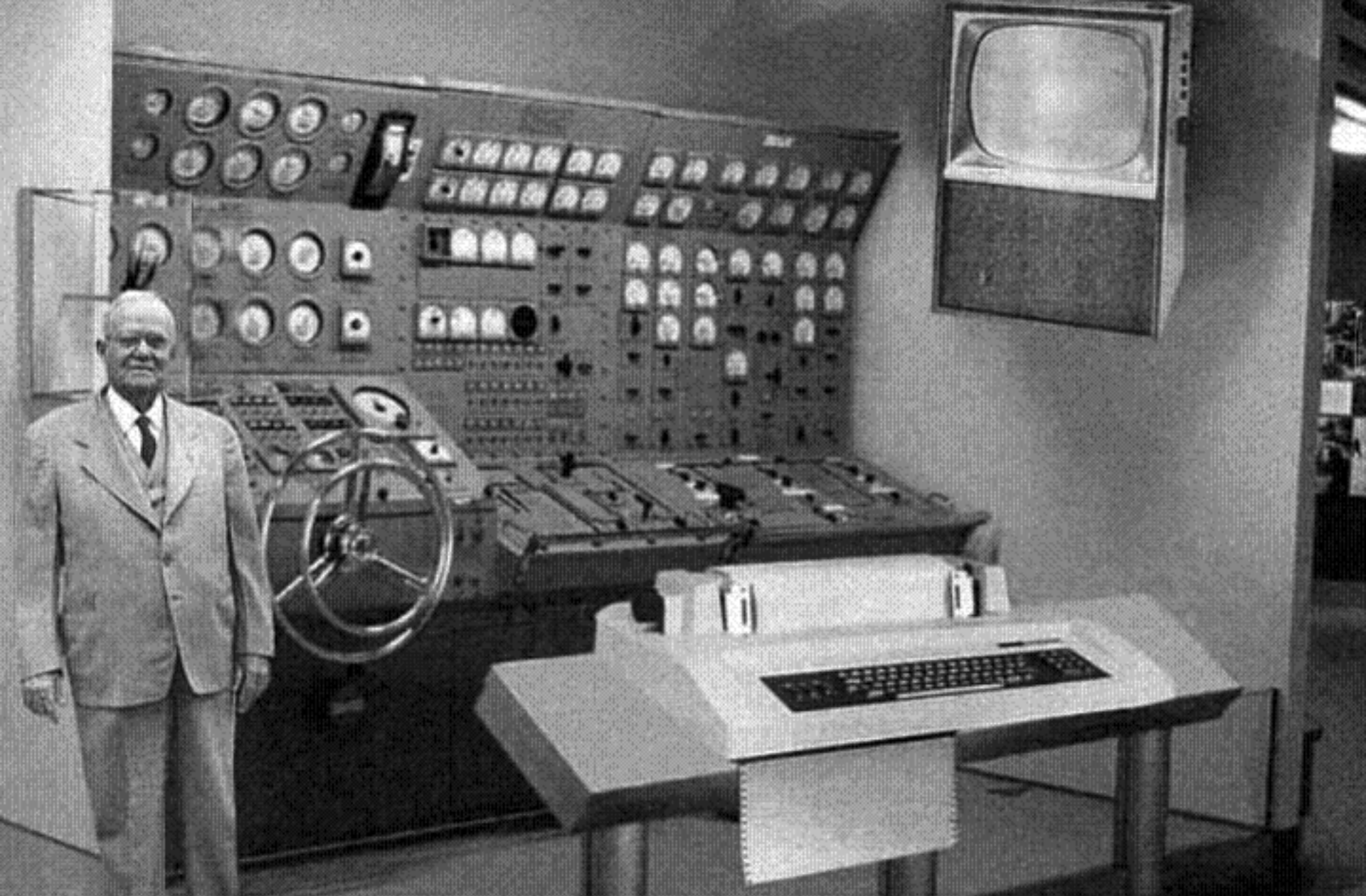
A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- Early mainframe stuff—IBM's and Univac's.
- God's gift to frustration—Analog & hybrid computers.
- Origins of the Internet—Arpanet and UCSB
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- Publish and perish. Too much of a good thing?
- Spread sheeting & plant profitability estimates.
- “What goes around, comes around:” Too much computing? => Re-focus on fundamentals.

Q: Did it all really happen this way?

A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- Early mainframe stuff—IBM's and Univac's.
- God's gift to frustration—Analog & hybrid computers.
- Origins of the Internet—Arpanet and UCSB.
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- Publish and perish. Too much of a good thing?
- Spread sheeting & plant profitability estimates.
- “What goes around, comes around:” Too much computing? => Re-focus on fundamentals.



Scientists from the RAND Corporation have created this model to illustrate how a "home computer" could look like in the year 2004. However the needed technology will not be economically feasible for the average home. Also the scientists readily admit that the computer will require not yet invented technology to actually work, but 50 years from now scientific progress is expected to solve these problems. With teletype interface and the Fortran language, the computer will be easy to use.



Q: Did it all really happen this way?

A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- Early mainframe stuff—IBM's and Univac's.
- God's gift to frustration—Analog & hybrid computers.
- Origins of the Internet—Arpanet and UCSB.
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- Publish and perish. Too much of a good thing?
- Spread sheeting & plant profitability estimates.
- “What goes around, comes around:” Too much computing? => Re-focus on fundamentals.



# Winter 1958—DuPont Textile Fibers

## Dacron Research Laboratory (Kinston, North Carolina)

- Bendix G-15 purchased by research laboratory. Learned to program the world's first personal computer. Key feature → no “operator” needed.
- Prob. Solved: A single polymer reactor (“finisher”) feeds multiple spinning machines from complex polymer supply manifold. Program minimized all residence times (time to degrade) and pressure drops (heat buildup) subject to constraints.
- Today, would be viewed as Pareto optimization with constraints, i.e., fiber production requirements.

**Bendix G-15 computer. Introduced in 1956 by the Bendix Corp, Computer Div. of Los Angeles, was about the size of a refrigerator, weighed 950 lb (450 kg). The base system, without peripherals, cost \$49,500. A working model cost around \$60,000.**



Q: Did it all really happen this way?

A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- **Early mainframe stuff—IBM's and Univac's.**
- God's gift to frustration—Analog & hybrid computers.
- Origins of the Internet—Arpanet and UCSB.
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- Publish and perish. Too much of a good thing?
- Spread sheeting & plant profitability estimates.
- “What goes around, comes around:” Too much computing? => Re-focus on fundamentals.

Fall 1957--IBM 650: Georgia Tech's first undergraduate student-accessible mainframe



## IBM 650 Characteristics

- 2000 words of signed-decimal drum memory.
- Used Bell Lab's "L1 compiler."
- 3 addresses: 1<sup>st</sup> (loc. where result is stored), (2<sup>nd</sup>: loc. of next executable command), 1<sup>st</sup> & 3<sup>rd</sup> were operands. User "responsible" for memory.
- Used in first undergrad heat transfer course to design multi-pass heat exchanger. (Without time to master logic commands, could only print out a ream of solutions and choose the best by inspection!!)

# Fall 1959—Technische Hochschule Stuttgart

Used Zuse Z22, built in 1956; Last working example is located at  
Zentrum für Kunst- und Medientechnologie in Karlsruhe



## The Zuse Z22

- First vacuum tube machine designed/constructed by Konrad Zuse about 1955 (earlier designs used mechanical relays).
- Z22 was programmed in machine code with 38 bit instruction words, consisting of 5 fields: the first 2 bits must always be 10, the next 5 bits contain a condition symbol, the next 13 bits contain an operation symbol, the next 5 bits contain a core memory address, the next 13 bits contain a drum memory address. A rudimentary compiler was available.
- Sat in on a fundamental computer science course for two semesters. Finally learned binary, hex, etc. representations, binary logic, and how to program in machine/assembly language *on a rational basis*.



## 1960 – 1964 -- Purdue and PhD research

- Fall 1960—Univac w/ Fortran compiler (Nirvana!)
- Nov 4, 1960 - From *Indiana Standard Gives Computer to Purdue*: “Standard Oil Company of Indiana has donated an electrical analog computer to Purdue University chemical engineering school. The computer will be used in a new process dynamics and control course for chemical engineering seniors.”
- 1963 - Coughanowr, D.R. & Koppel, L.B.; *Process Analysis and Control*, McGraw-Hill

Q: Did it all really happen this way?

A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- Early mainframe stuff—IBM's and Univac's.
- God's gift to frustration—Analog & hybrid computers.
- Origins of the Internet—Arpanet and UCSB.
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- Publish and perish. Too much of a good thing?
- Spread sheeting & plant profitability estimates.
- “What goes around, comes around:” Too much computing?  
=> Re-focus on fundamentals.

# Summer 1961--Dow Chemical Company, Midland, MI Process Fundamentals Laboratory



Summer Assignment: Analyze a large set of laboratory reactor data and develop a kinetic model for dehydrogenation of ethylbenzene to styrene

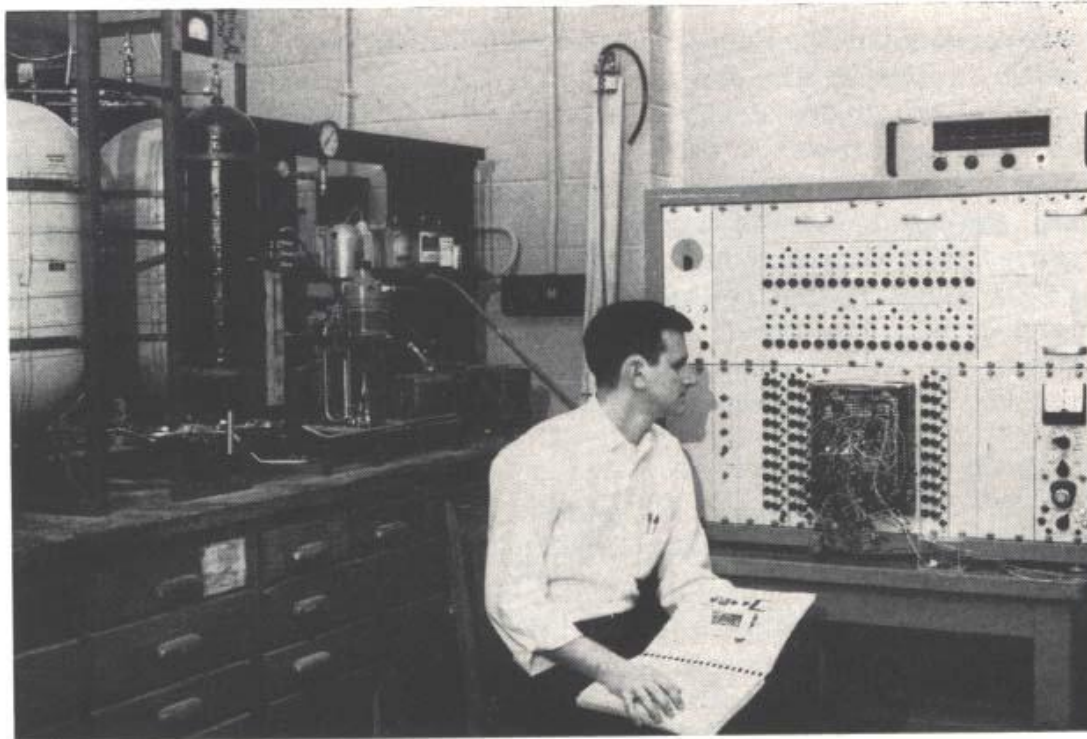


Side reactions produce benzene and toluene.

Procedure:

- \* Use analog computer to simulate reactor (three reaction equations, etc.)
- \* “Plot” lab data on oscilloscope screen with crayon,
- \* Fit kinetic parameters while running computer in “repetitive-operation” mode (1000 solutions/sec.)
- \* 160 hours to program analog; 1 hour to fit data.

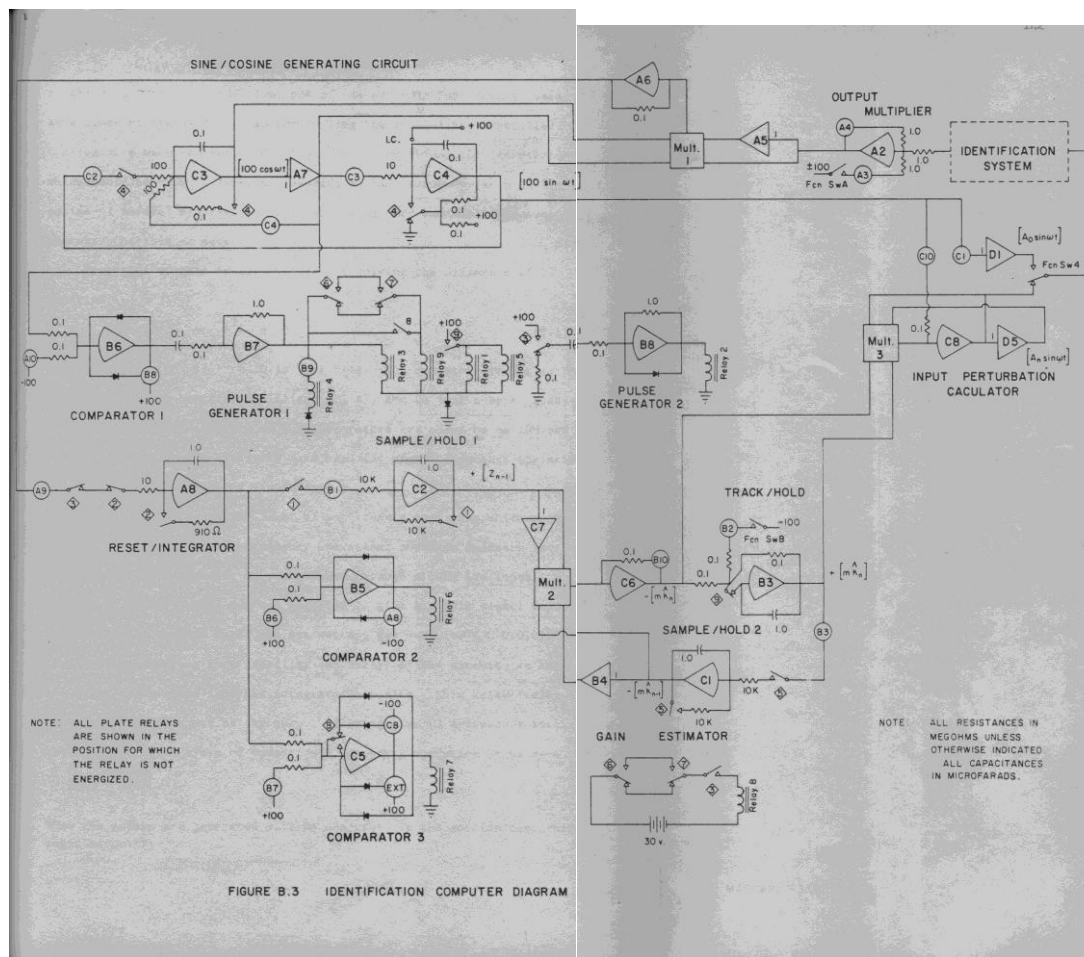
1961/1964—Purdue: PhD laboratory research on Adaptive Control of pH using department's “new” Applied Dynamics AD-32 analog computer



**A study of adaptive control of pH.**

# Schematic Diagram of Gain Identifier:

Required 8 relays to implement digital logic to estimate gain of pH system once per three minutes





# 1965-86 -- UCSB Arts Building

Temporary (!!!) home of Department of Chemical Engineering





# Key Events in Development of Department

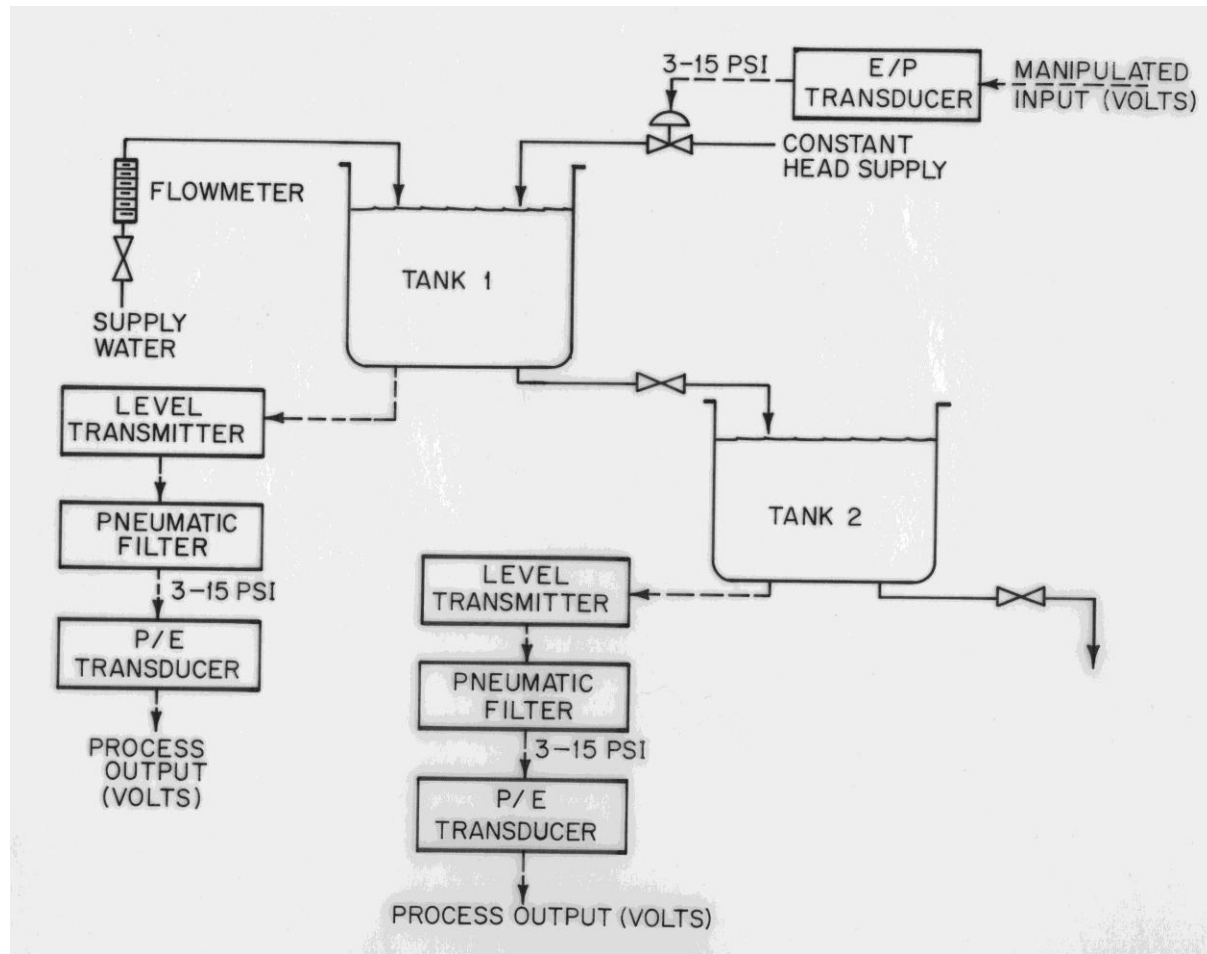
- 1965 First faculty member hired: Bob Rinker/Caltech.
- 1966 Founding department chair hired (Jack Myers/Purdue); DAM arrives.
- 1967 Final two (of six) founding ChE faculty hired. Undergrad curriculum completed and approved by Academic Senate.
- 1968 First graduating class (six students). MS program in ChE approved.
- 1969 PhD program in ChE approved at UC Systemwide level.
- 1970 ChE Department first MS degree granted.
- 1971 College of Engineering first PhD granted in EE Department
- 1973 ChE Department first PhD granted (Joe Ault: Dissertation: *Modeling of Polycondensation Systems*)
- 1976 Dale Seborg arrives. First ChE faculty hired since 1967.
- 1978 College of Engineering first PhD granted to female: (Sandra Harris)
- 1986 Department moves into new building and COE adopts focus on interdisciplinary hires as cornerstone of program building efforts.
- 2010 Recent NRC research evaluations: validation of early approach(?).

1967 – First undergraduate process control experiments built--  
Two-tank liquid level process and two-tank heating process



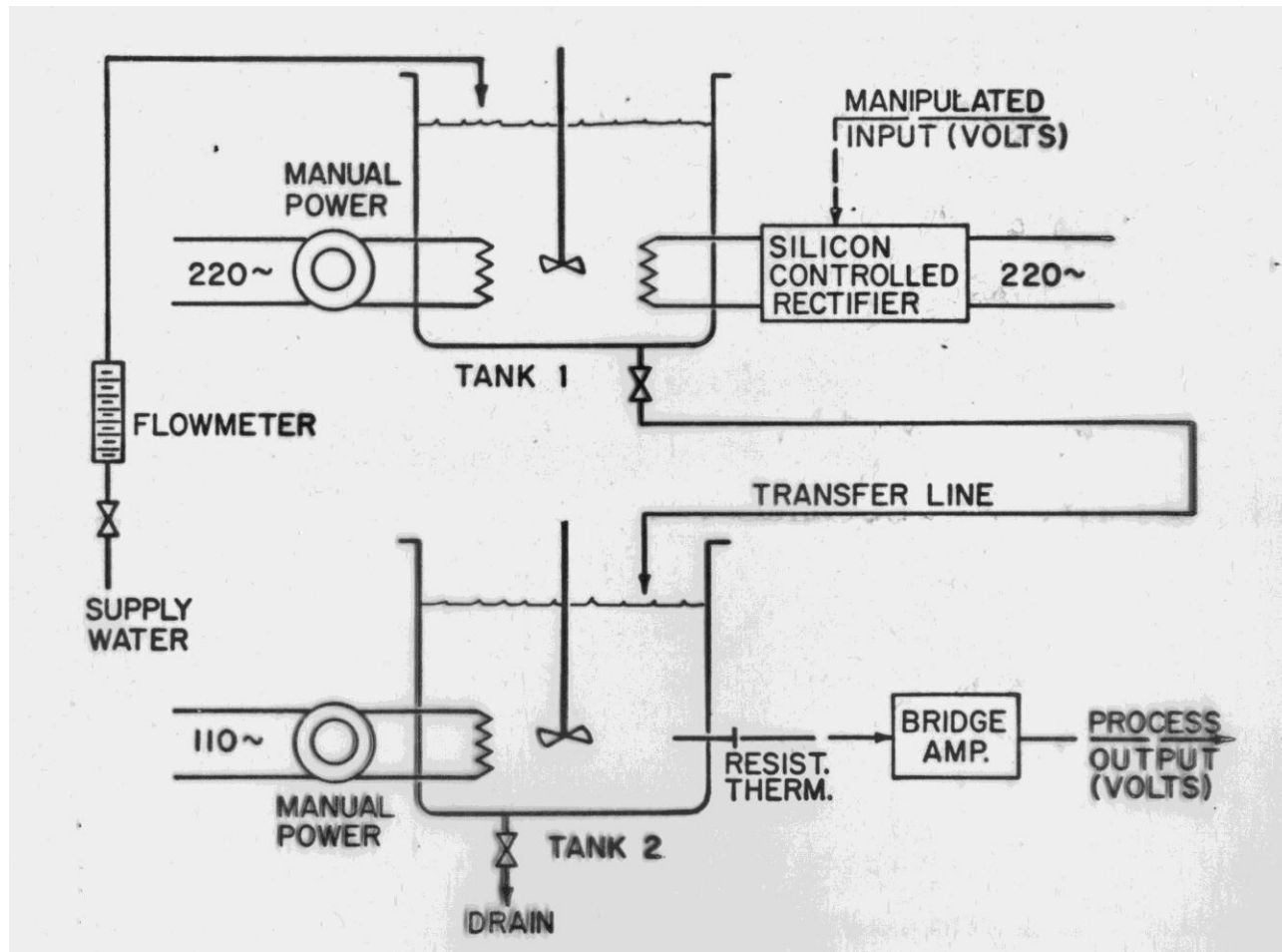
# Two-Tank Liquid Level Process

Note: Time constant of each tank approx. 3 minutes



# Two-Vessel Stirred-Tank Process

Note: Time constant of each tank approx. 3 minutes

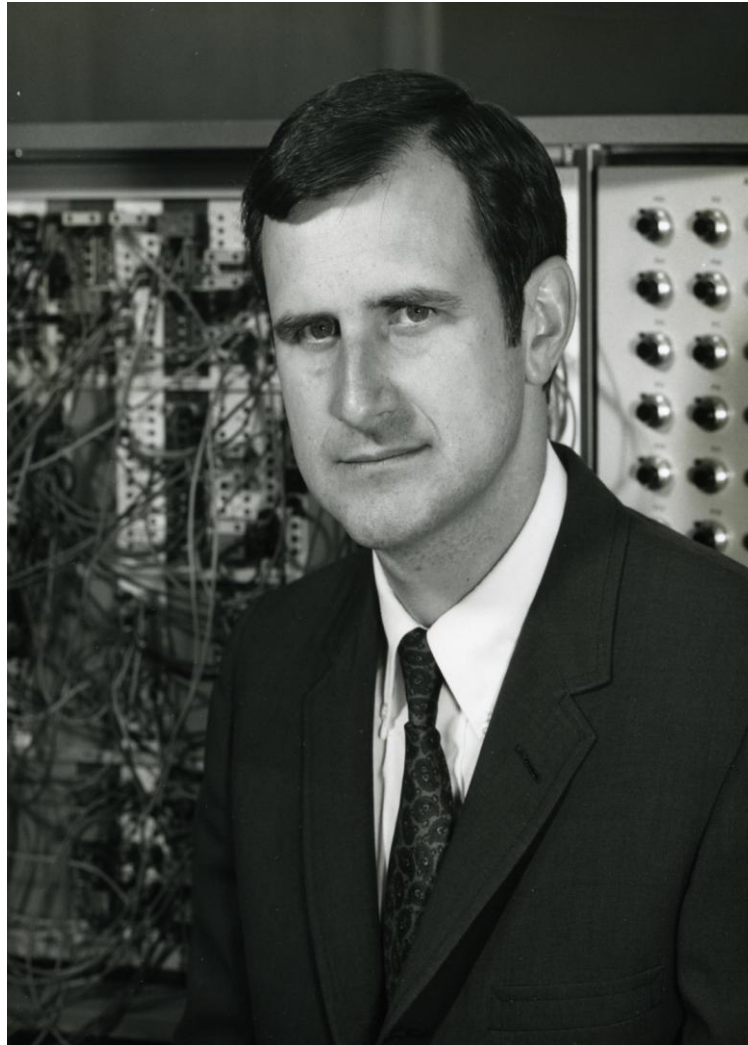


Summer 1967—UC Santa Barbara

A used Electronic Associates TR-48 for new Process Control laboratory  
(~\$15,000 blown of \$25,000 start-up fund)



A green assistant professor with his pride-and-joy ...  
obviously didn't have a clue what was coming.





Q: Did it all really happen this way?

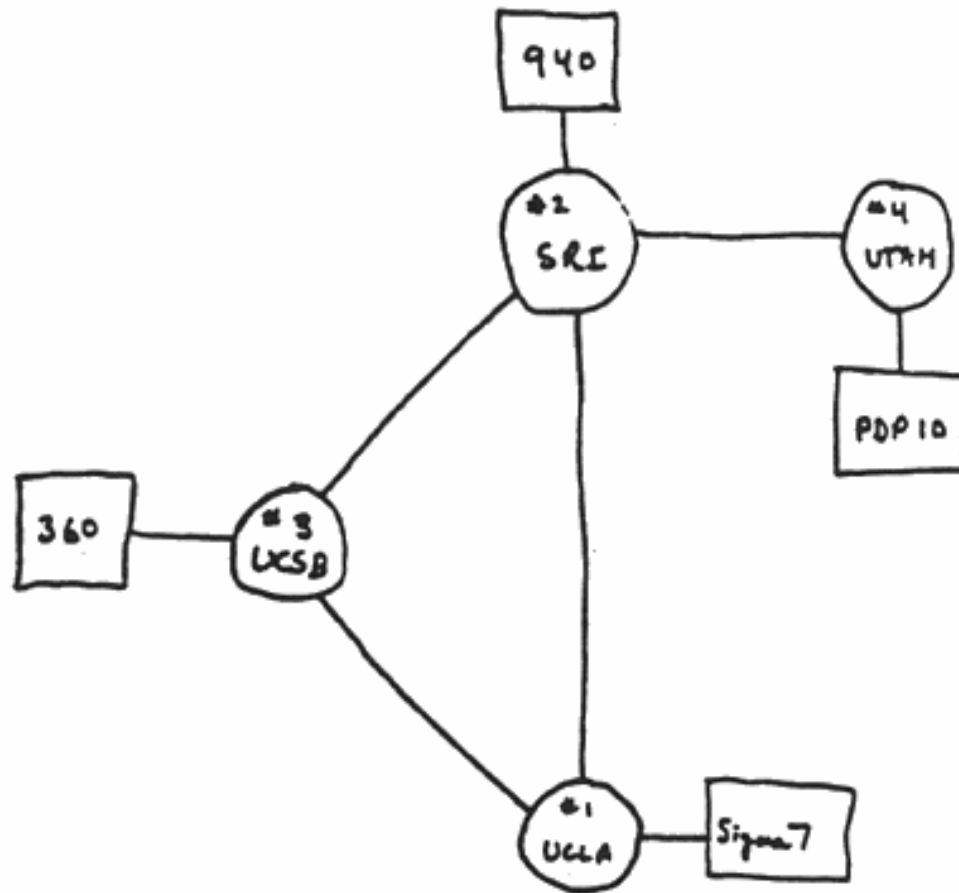
A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- Early mainframe stuff—IBM's and Univac's.
- God's gift to frustration—Analog & hybrid computers.
- **Origins of the Internet—Arpanet and UCSB.**
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- Publish and perish. Too much of a good thing?
- Spread sheeting & plant profitability estimates.
- “What goes around, comes around:” Too much computing? => Re-focus on fundamentals.



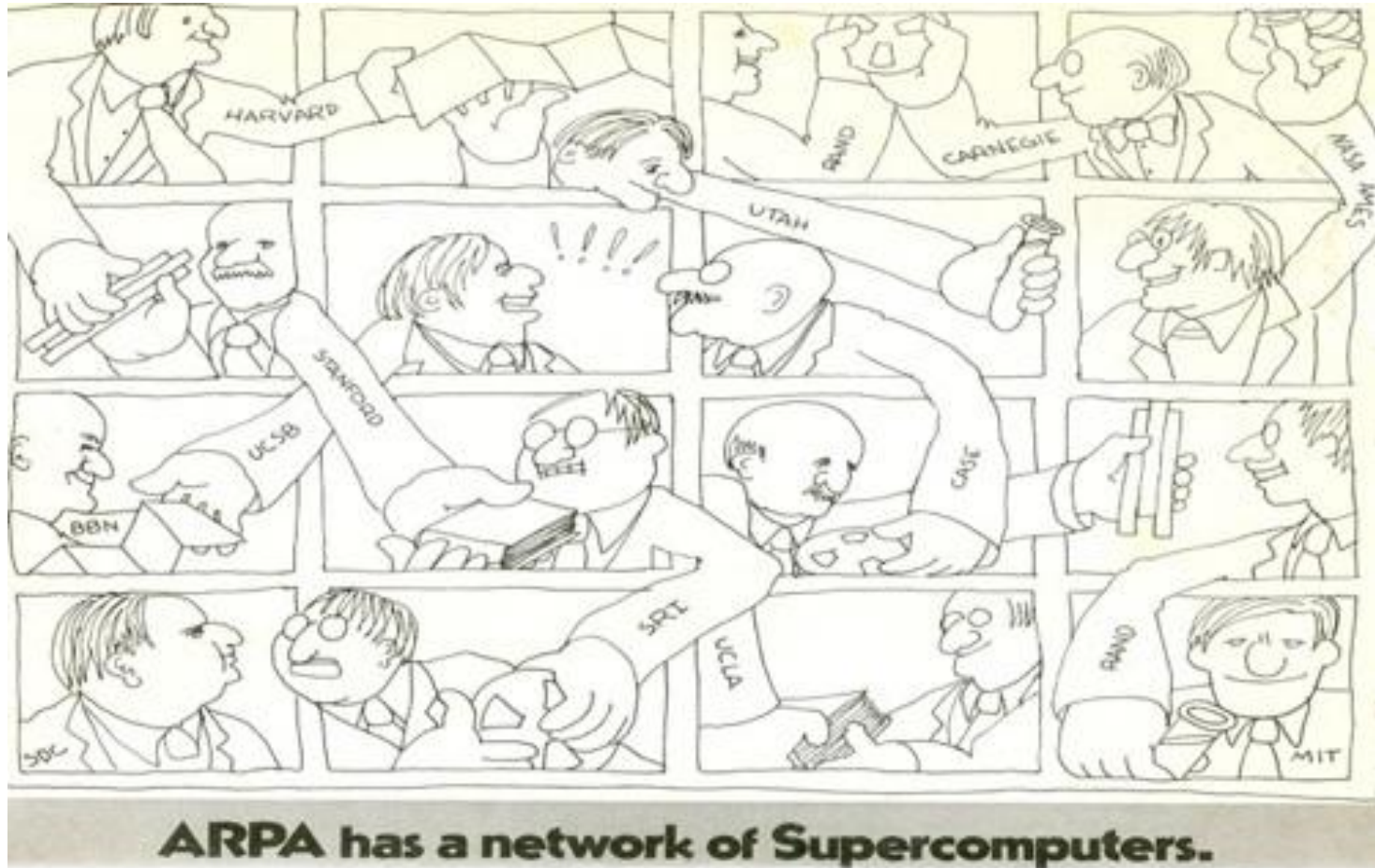
# 1969 The Arpanet and UCSB

## First four nodes of the Internet!



# Expansion of the Arpanet to Midwest and East Coast Locations

“Packet-Switching” on the Internet (Note: All white males!)

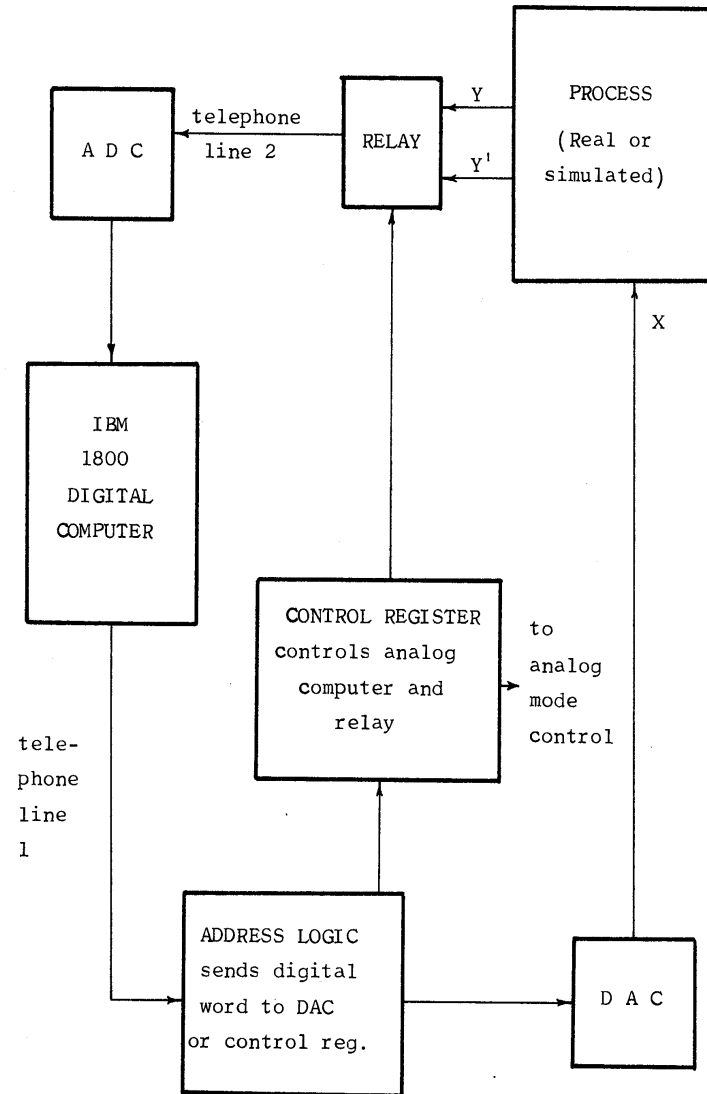


Q: Did it all really happen this way?

A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- Early mainframe stuff—IBM's and Univac's.
- God's gift to frustration—Analog & hybrid computers.
- Origins of the Internet—Arpanet and UCSB.
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- Publish and perish. Too much of a good thing?
- Spread sheeting & plant profitability estimates.
- “What goes around, comes around:” Too much computing? => Re-focus on fundamentals.

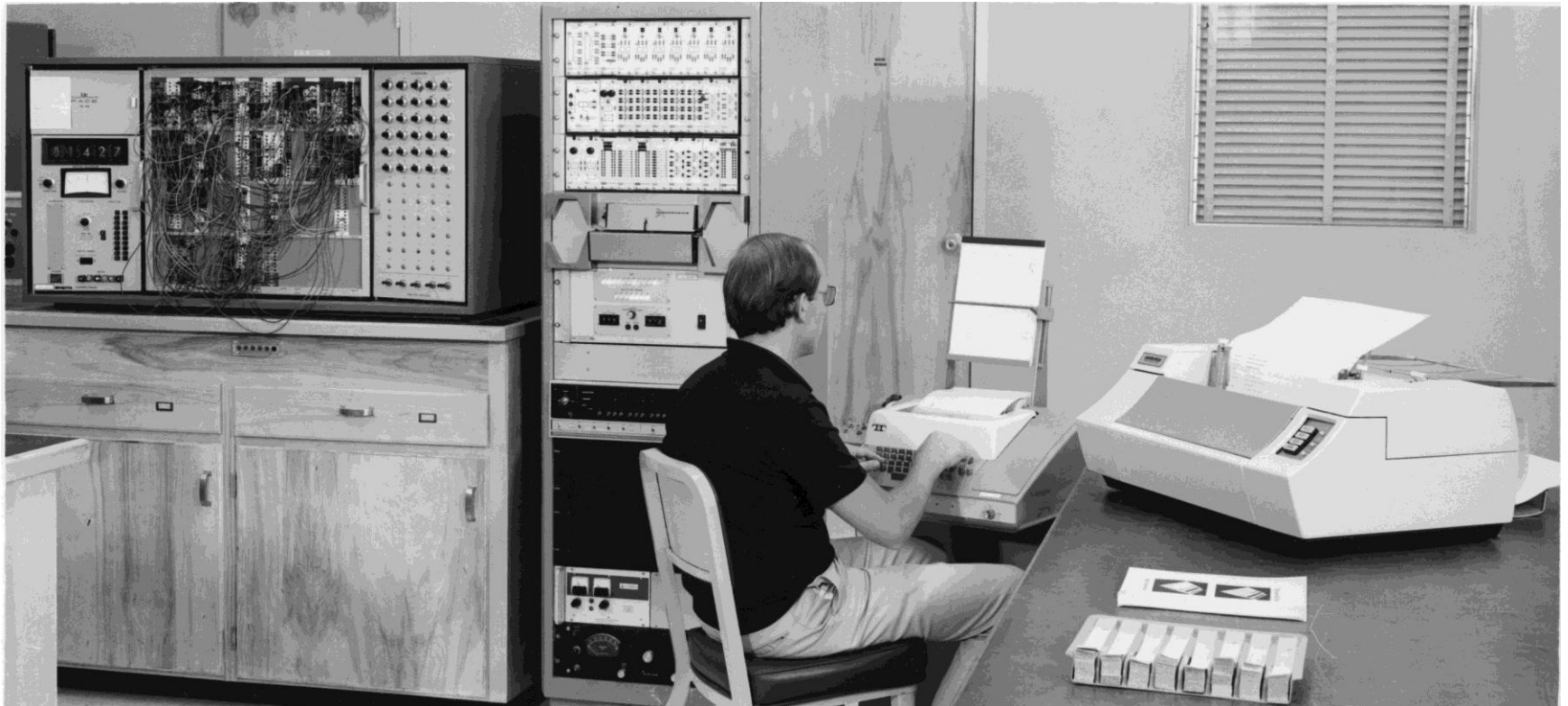
# 1968-First experiments in on-line process identification using remote IBM 1800 in Electrical Engineering (1/2 mi. across campus)



1970-First true digital computer in Process Control Laboratory:  
Data General Corp. NOVA [16-bit word/4 accumulators/8,000  
bytes of core memory]

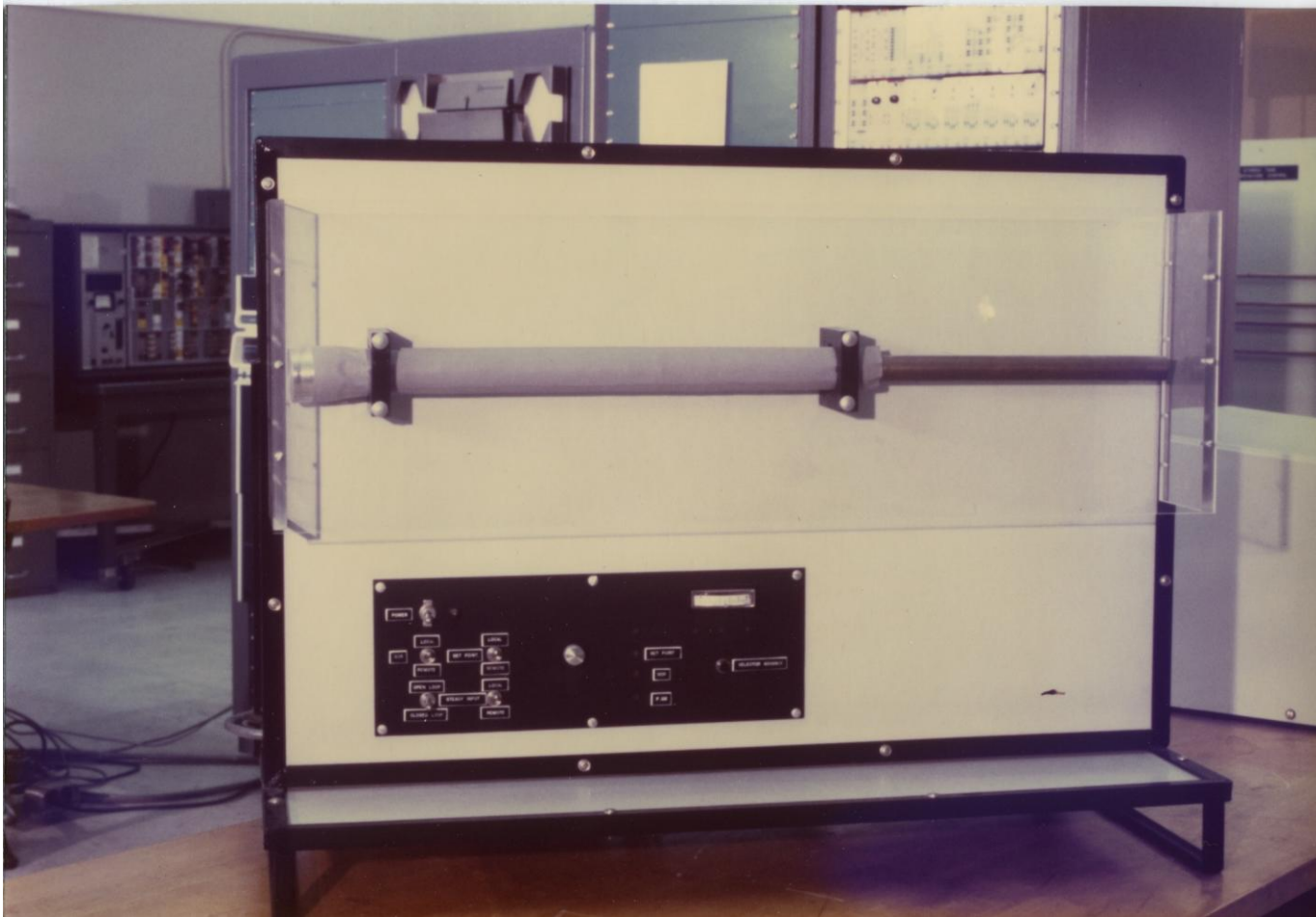


1972-Electronic binary logic modules and NOVA computer  
interfaced to Analog Computer and PC Lab  
[Note: Teletype for 10 char/sec input; H/S Paper tape reader]



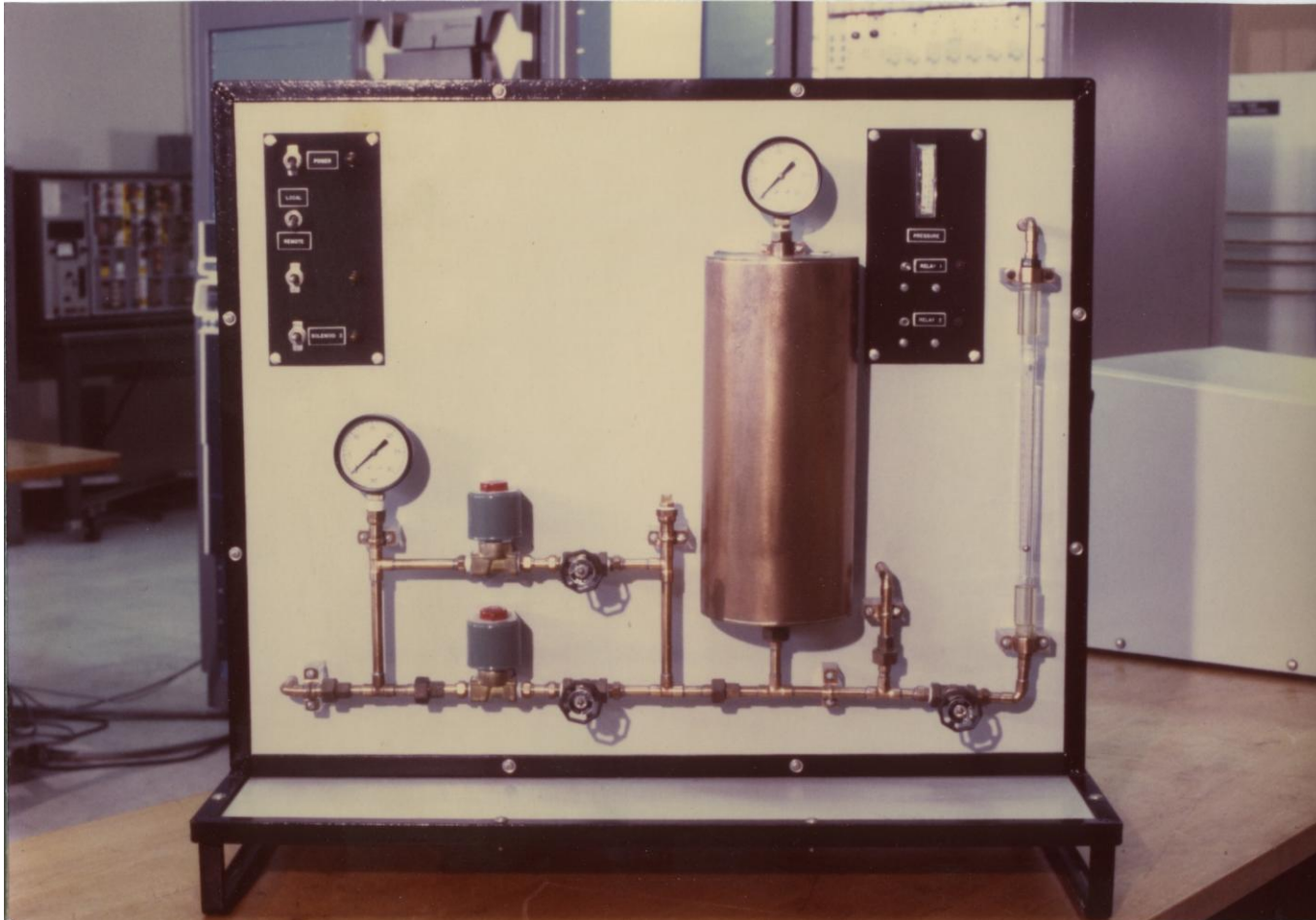


1974-Added new experiment -- “Heated Bar”—to help teach a new three-course real-time computing sequence





Many EE and CS students signed up for RT courses.  
Built a new experiment they could relate to—RC  
“circuit.”



1975-New 80-column Hollerith card reader added (real progress!) Soon replaced TTY with CRT terminals; added SuperNova and Eclipse Computers)



Process Control and Real-Time Laboratories needed a quick method for interfacing individual experiments.  
[Note separate patch panel for each separate experiment.]

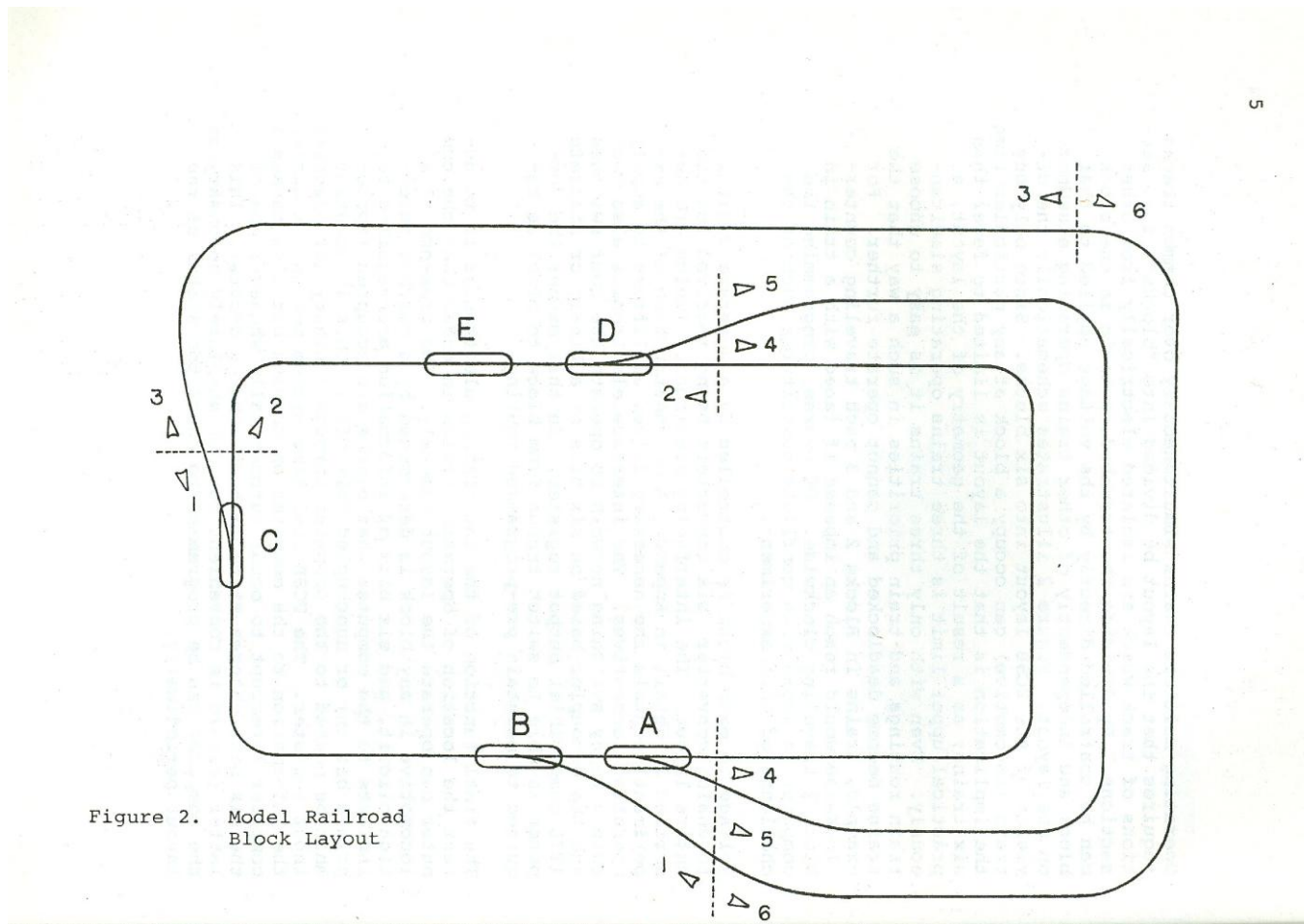




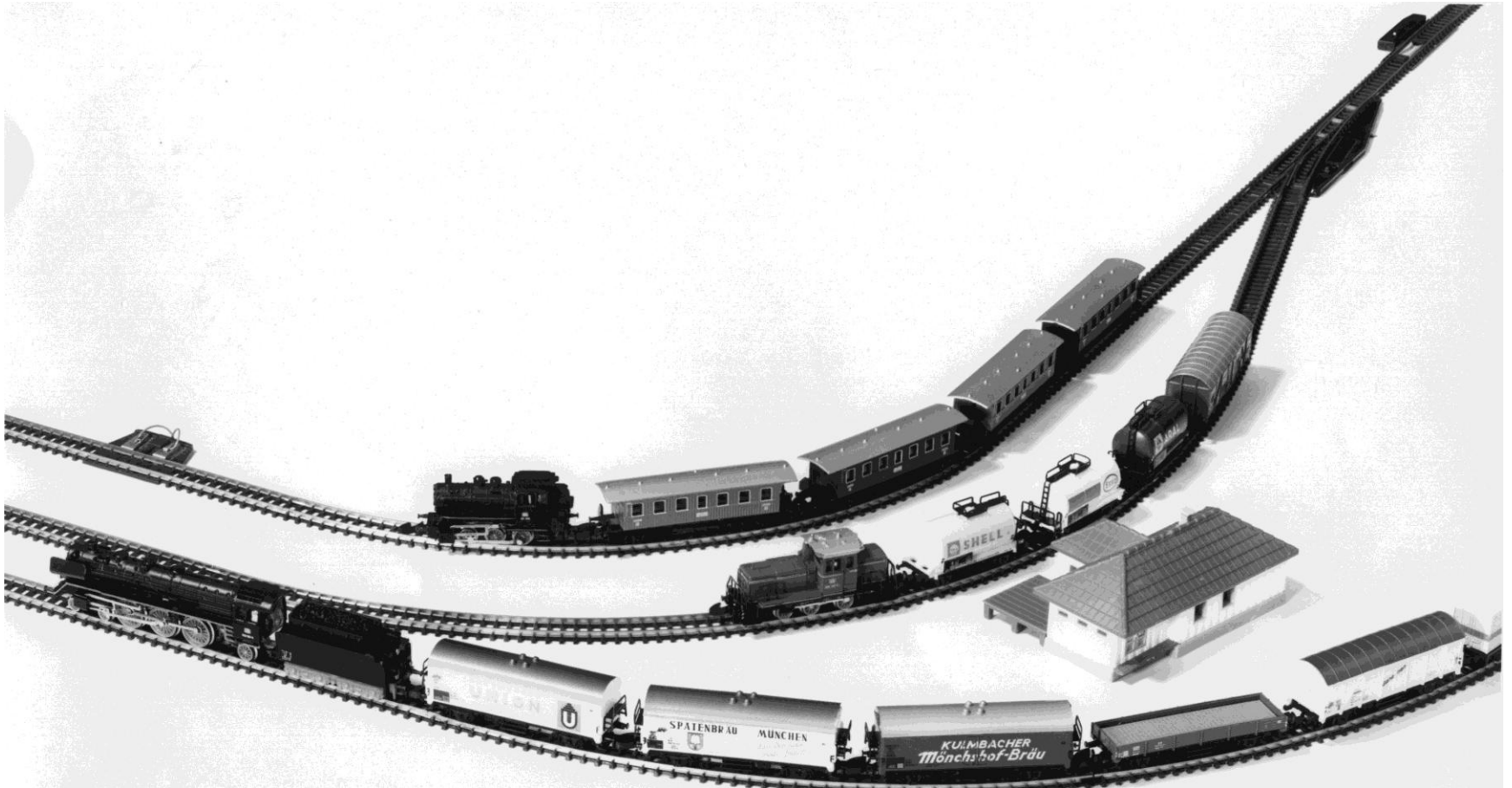
1975--All new computers operational. Two new PC  
experiments built (L/L #2 and STH #2)  
[Anyone recognize who is looking over student's shoulder?]



1975--What's missing? A fun, easy-to-understand experiment that has unmistakable "failure mode behavior" --> Märklin z-gauge Model R/R w/ 6 controllable "blocks" and 4 switches.

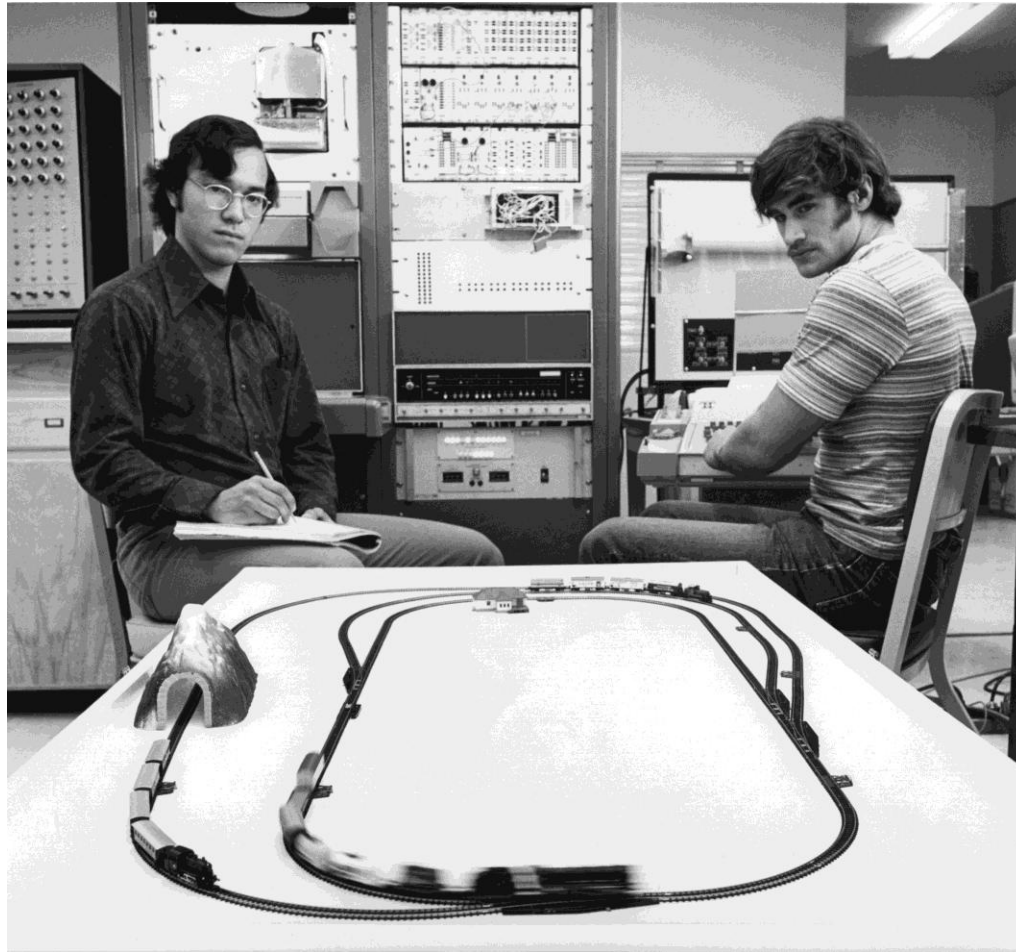


Model Railroad--Things working the way they are supposed to work: one train proceeding; two trains slowed or halted.

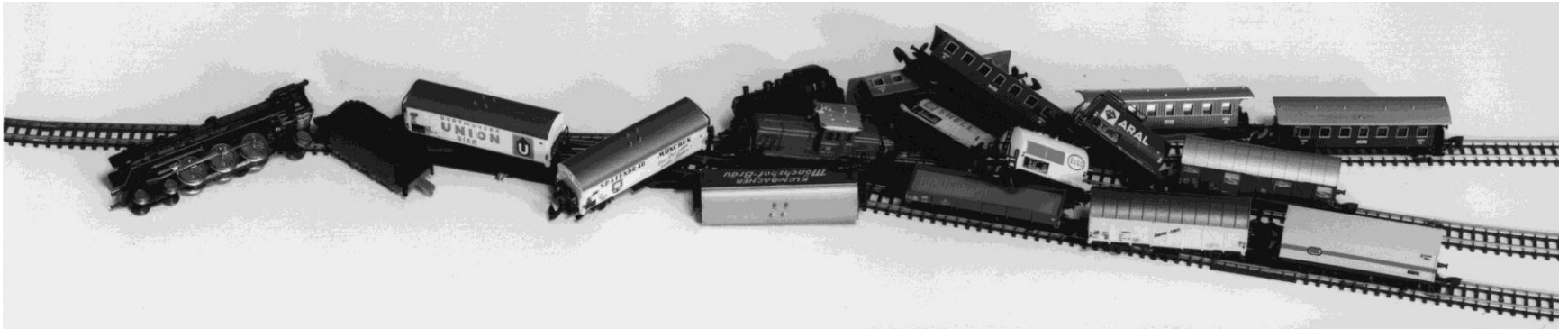




Undergraduates running three train  
over preselected repetitive routes.  
(Note: Ed McNeil (Aspentech's MPC cat cracker expert.)



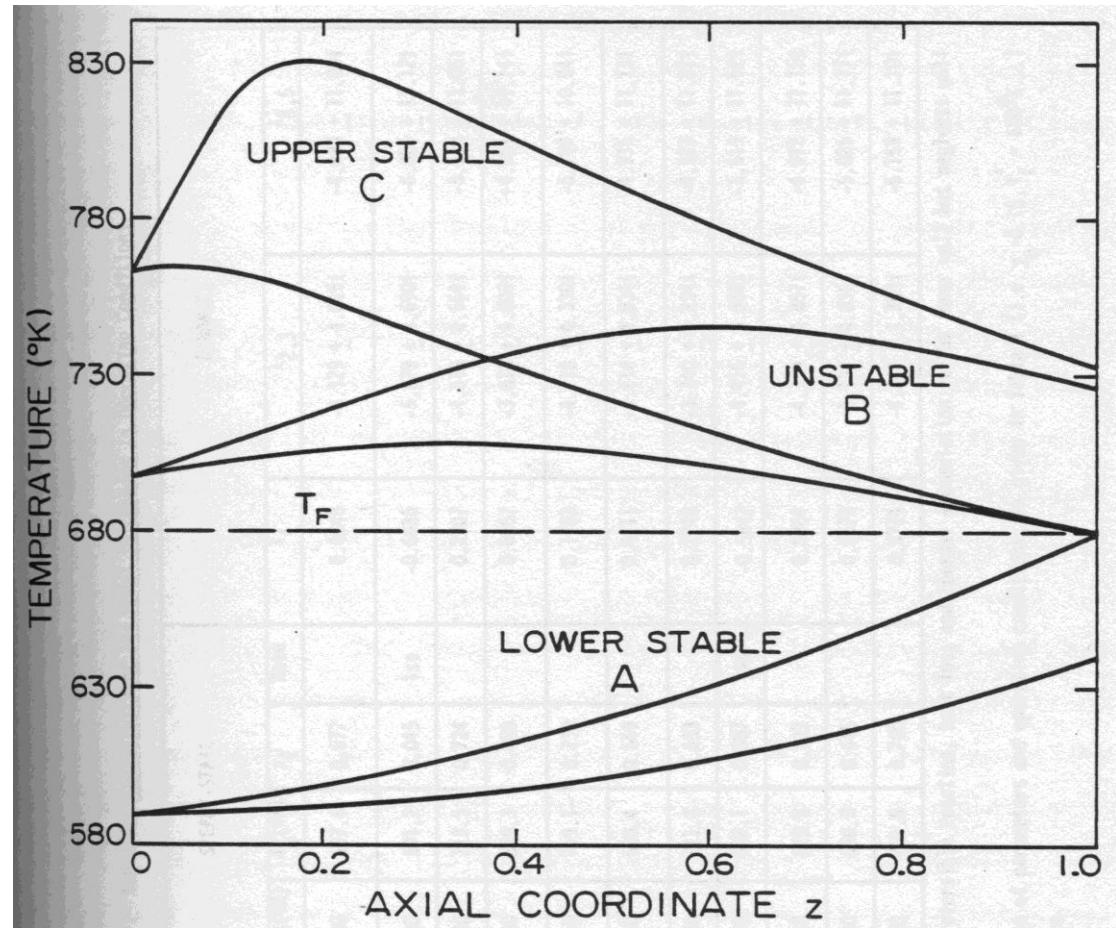
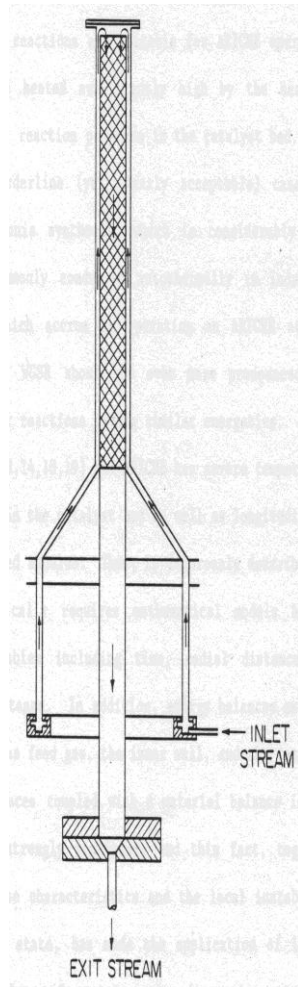
Model Railroad: Woops! Programming error.  
Students back to the drawing board!!



1976-Dale Seborg joins UCSB faculty and things really take off.  
Pictured: Orville Sandall, Gene Lucas, Dale Seborg, and author.  
Question: Who was undercover “narc” on the side?

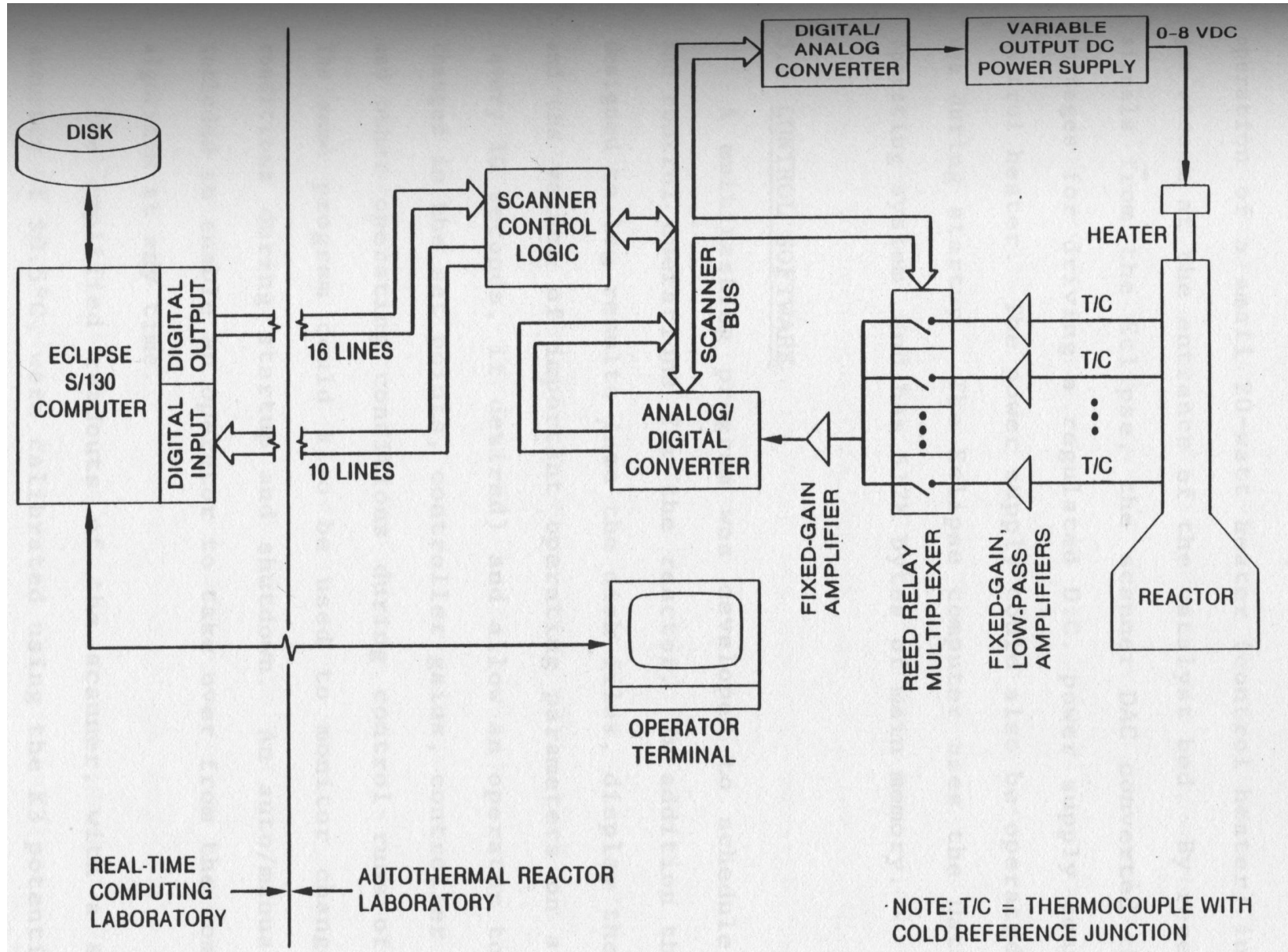


1980-First control of an unstable distributed chemical process:  
Autothermal, counterflow reactor running exothermic reaction  
(water gas shift:  $\text{CH}_4 + \text{CO}_2 \rightleftharpoons 2 \text{CO} + 2 \text{H}_2$ )





# Control of Unstable Autothermal Tubular Reactor



Q: Did it all really happen this way?

A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- Early mainframe stuff—IBM's and Univac's.
- God's gift to frustration—Analog & hybrid computers.
- Origins of the Internet—Arpanet and UCSB.
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- **Publish and perish. Too much of a good thing? More?**
- Spread sheeting & plant profitability estimates.
- “What goes around, comes around:” Too much computing? Re-focus on analytical solutions?



# 1977-79--CACHÉ Monographs in Real-Time Computing,

D.A.Mellichamp, Editor

## Contents of Eight Monographs:

### Part I. AN INTRODUCTION TO REAL-TIME COMPUTING

1. Digital Computing and Real-Time Digital Computing, *Duncan A. Mellichamp* / 3
2. The Structure of Real-Time Systems, *Duncan A. Mellichamp* / 12
3. An Overview of Real-Time Programming, *Duncan A. Mellichamp* / 33

### Part II. PROCESSES, MEASUREMENTS, AND SIGNAL PROCESSING

4. Processes and Representative Applications, *Thomas F. Edgar* / 55
5. Measurements, Transmission, and Signal Processing, *Joseph D. Wright* / 80

### Part III. INTRODUCTION TO DIGITAL ARITHMETIC AND HARDWARE

6. Representation of Information in a Digital Computer, *D. Grant Fisher* / 113
7. Digital (Binary) Logic and Hardware, *George P. Engelberg* and *James A. Howard* / 139

### Part IV. REAL-TIME DIGITAL SYSTEMS ARCHITECTURE

8. Digital Computer Architecture, *George P. Engelberg*, *William R. Hughes*, and *James A. Howard* / 165
9. Peripheral Devices and Data Communications, *Walter G. Rudd* / 198
10. Digital Computer/Process Interfacing, *William R. Hughes* / 235

### Part V. REAL-TIME SYSTEMS SOFTWARE

11. Assembly Language Programming, *D. Grant Fisher* / 281
12. Utility and Systems Software, *James Wm. White* and *Joseph D. Wright* / 312
13. Real-Time Operating Systems and Multitask Programming, *Joseph D. Wright* and *James Wm. White* / 345

### Part VI. REAL-TIME APPLICATIONS SOFTWARE

14. Real-Time BASIC, *Duncan A. Mellichamp* / 381
15. Real-Time FORTRAN, *James Wm. White* / 412
16. Control-Oriented Languages (Table-Driven Software), *Cecil L. Smith* / 448

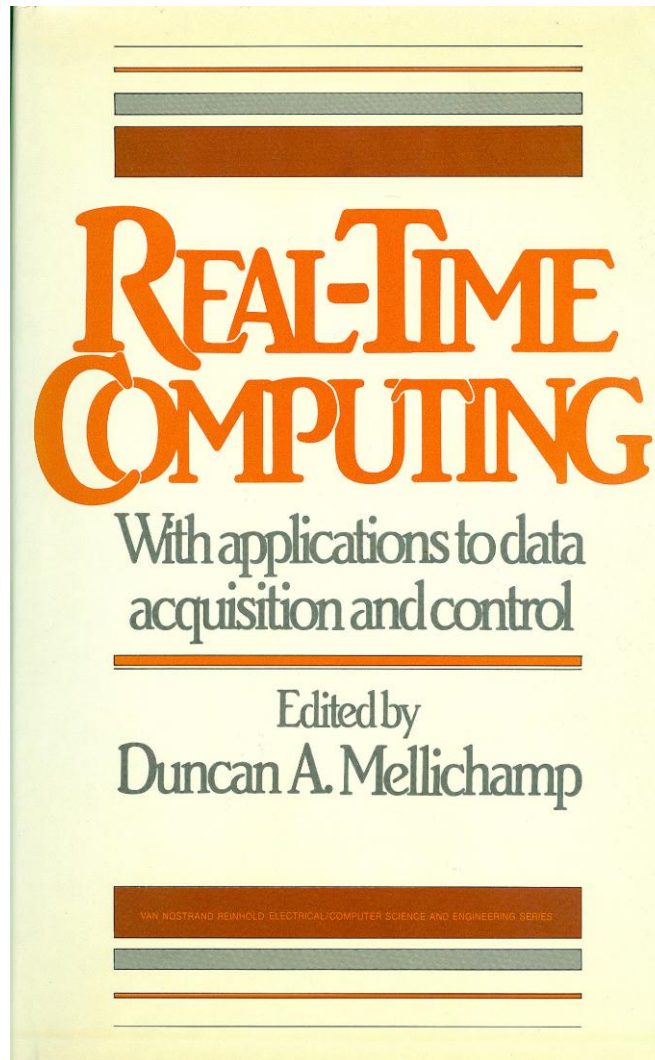
### Part VII. MANAGEMENT OF REAL-TIME COMPUTING FACILITIES

17. System Justification, Selection, and Installation, *Cecil L. Smith* / 463
18. System Operations, Management, and Program Documentation, *David E. Clough* / 475

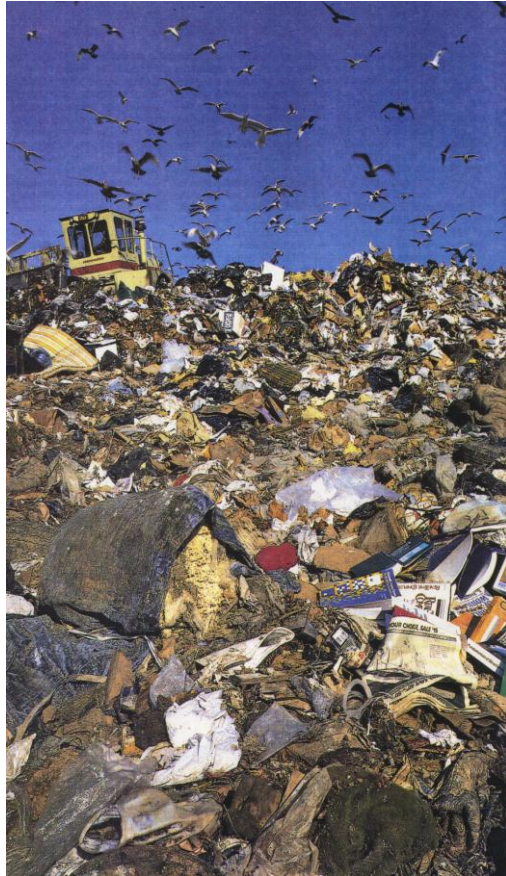
### Part VIII. PROCESS ANALYSIS, DATA ACQUISITION, AND CONTROL ALGORITHMS

19. Process Analysis and Description, *Thomas F. Edgar* / 491
20. Digital Computer Control and Signal Processing Algorithms, *Joseph D. Wright* and *Thomas F. Edgar* / 519

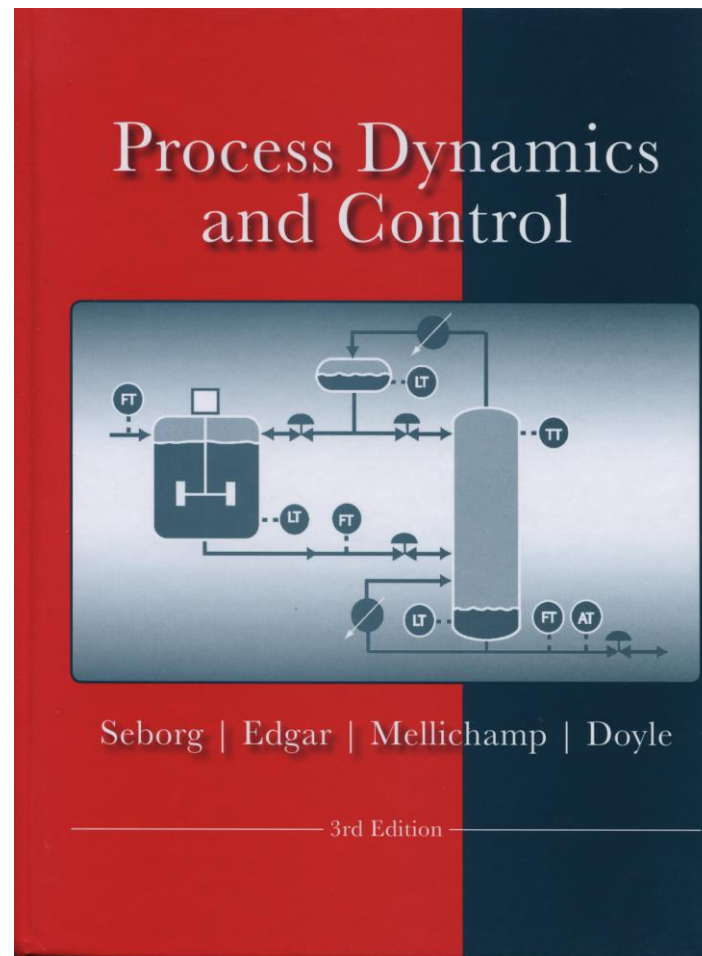
1983--*Real-Time Computing*, D.A.Mellichamp, Ed.,  
Van Nostrand Reinhold



~1986–*American Scientist* published article on seriousness of national waste disposal problem.



1980-2010: The *Perpetual Book Project* with co-authors  
Dale Seborg, Tom Edgar, and (w/ 3<sup>rd</sup> Ed.) Frank Doyle



Q: Did it all really happen this way?

A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- Early mainframe stuff—IBM's and Univac's.
- God's gift to frustration—Analog & hybrid computers.
- Origins of the Internet—Arpanet and UCSB.
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- Publish and perish. Too much of a good thing?
- **Spread sheeting & plant profitability estimates.**
- “What goes around, comes around:” Too much computing? => Re-focus on fundamentals.



# Discounted Cash Flow Calculations – Adapted from Jim Douglas, Ex. 2.5-1

		Discount Rate (DR) =		20%	Tax_Rate		48%
Fixed Capital =							
FC	100,000,000	a-3	0		b_1	0.60	
WC	20,000,000	a-2	0.15		b_2	0.90	
SU	10,000,000	a-1	0.35		b_3	0.95	
SV	3,000,000	a0	0.5		Profit_BT	66,770,000	
					ROI_BT	51.4%	

Year		Cash Flows	Discount Factor	Present Value
-3	Investment Capital	0	1.7280	0
-2		-15,000,000	1.4400	-21,600,000
-1		-35,000,000	1.2000	-42,000,000
0		-50,000,000	1.0000	-50,000,000
	Working Capital	-20,000,000	1.0000	-20,000,000
	Start-Up Capital	-10,000,000	1.0000	-10,000,000
	PV of Investments (=Total Capitalized Investment)			-143,600,000
1		26,112,240	0.8333	21,760,200
2		36,528,360	0.6944	25,366,917
3		38,264,380	0.5787	22,143,738
4		40,000,400	0.4823	19,290,316
5		40,000,400	0.4019	16,075,264
6		40,000,400	0.3349	13,396,053
7		40,000,400	0.2791	11,163,378
8		40,000,400	0.2326	9,302,815
9		40,000,400	0.1938	7,752,346
10		40,000,400	0.1615	6,460,288
	Working Capital	20,000,000	0.1615	3,230,112
	Salvage Value	3,000,000	0.1615	484,517
	PV of Operations			156,425,942

NPV(0)=Sum Present Values (Investment Plus Operations) 12,825,942



An analytical expression can be derived for the  $NPV_0$ . It appears to be highly nonlinear, but ...

$NPV_0$  at start of operations period can be expressed analytically as:

$$NPV_0 = TR_c \left[ (Profit_{BT}) \left[ b_{+1}(1+DR)^{-1} + b_{+2}(1+DR)^{-2} + b_{+3}(1+DR)^{-3} + \dots + b_{+10}(1+DR)^{-10} \right] \right. \\ \left. + 0.1TR (FC + SU) \left[ (1+DR)^{-1} + (1+DR)^{-2} + (1+DR)^{-3} + \dots + (1+DR)^{-10} \right] \right. \\ \left. - TCI + \left[ TR_c (SU + SV) \right] (1+DR)^{-10} \right]$$

where  $Profit_{BT} = Revenues - Operating Costs$

$$TCI = a_{-3}FC(1+DR)^{+3} + a_{-2}FC(1+DR)^{+2} + a_{-1}FC(1+DR)^{+1} + a_0FC + WC + SU$$

$DR = Discount Rate$ ;  $TR = Tax Rate$ ;  $TR_c = Complementary Tax Rate, 1 - TR$ ;

On further analysis the expression for  $NPV_0$  turns out to have a trivially linear form(!!):

$$NPV_{proj} = a Profit_{BT} + b Fixed\_Capital \quad (1)$$

where  $NPV_{proj}$  is  $NPV_0$  discounted back to the date of project approval

Also, a “normalized”  $NPV$  quantity can be defined

$$NPV_{\%} = c NPV_{proj} / [(Fixed\_Capital)(Project\_Lifetime)] \quad (2)$$

It can be shown similarly that  $NPV_{\%}$  is linearly related to  $ROI_{BT}$ .

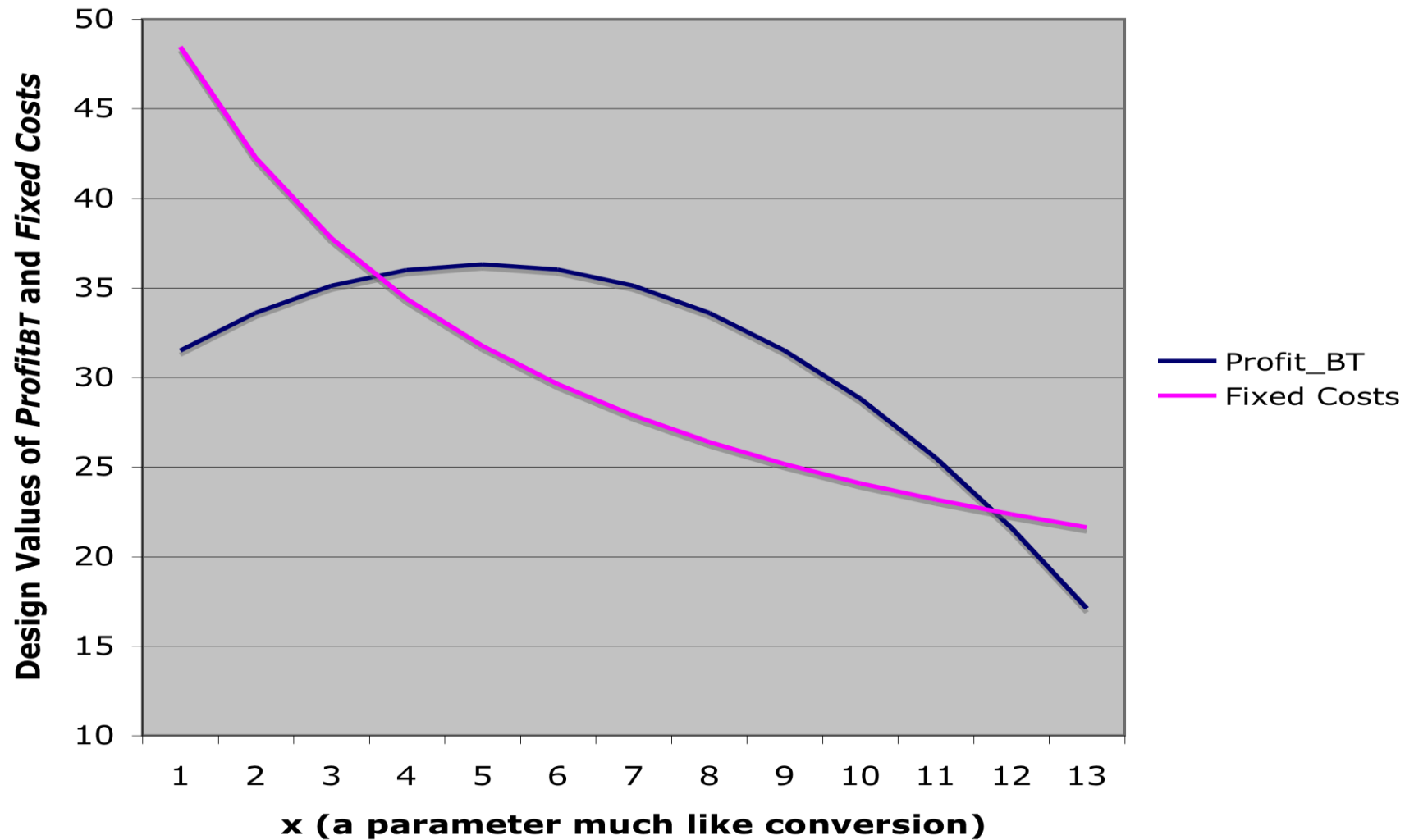
The surprising results are that the DCF spreadsheet can be eliminated and an equivalent linear model employed. Thus, it is trivially easy to evaluate a project's projected profitability.

Equations (1) and (2) plus several additional *linear expressions* replace an entire DCF spreadsheet!

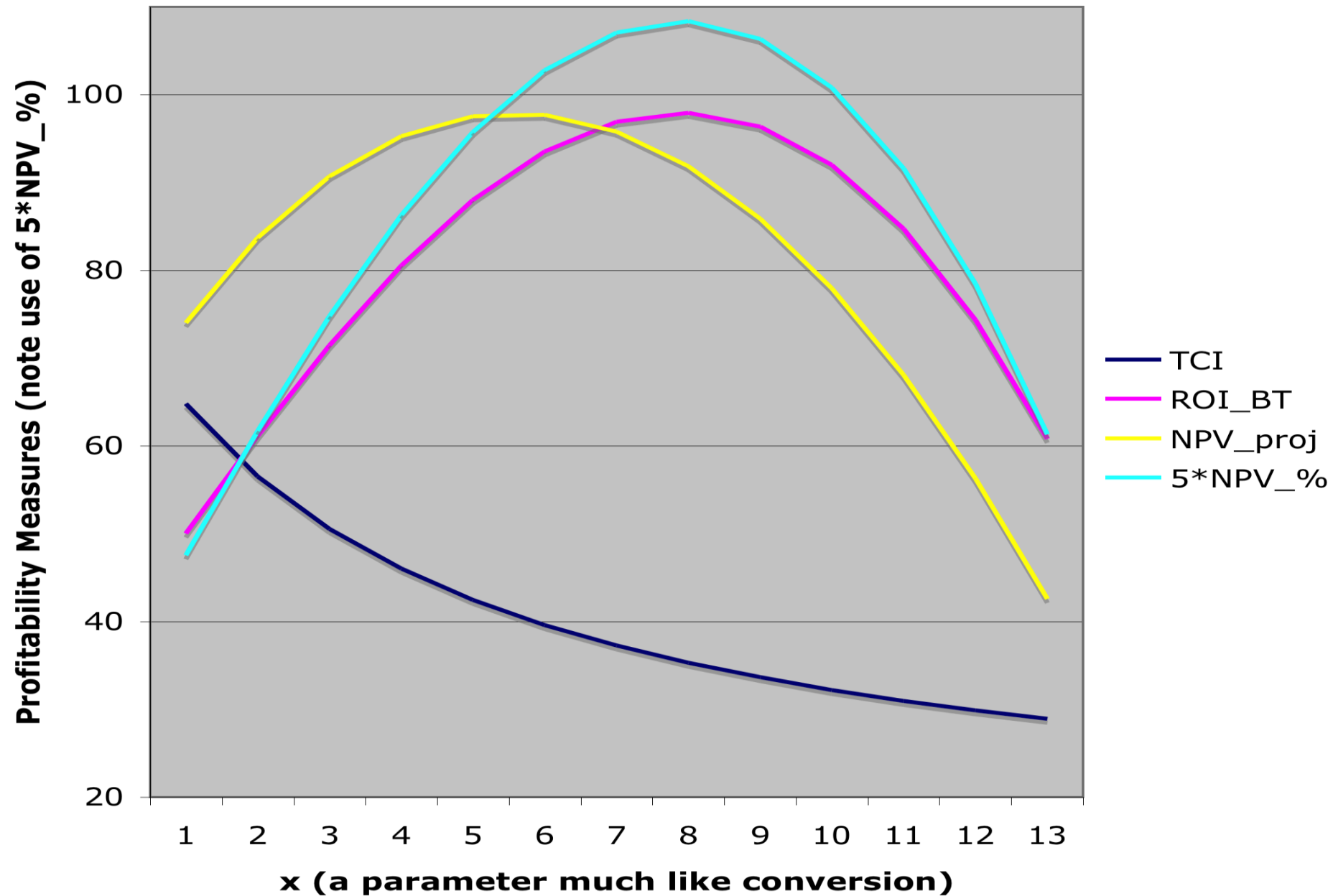
The designer can choose to optimize:

- a)  $NPV_{proj}$  (= prospective value added to company's net worth if the project is undertaken), or
- b)  $ROI_{BT} \propto NPV_{\%}$  (= rate of return on capital put at risk if the project is undertaken), or
- c) any arbitrary function of  $NPV_{proj}$  and  $ROI_{BT}$ .

**Figure 5.1 Design Values:  $Profit_{BT}$  and Fixed Costs**  
**as Fcns of Design Parameter**



**Figure 5.2 Optimization of Plant Profitability**



## Conclusions arising from recent personal research:

- Now have a simple linear model that can be used intuitively.
- $NPV_{\%}$  is inversely related to “risk.” It should be maximized in order to recover capital at risk quickly. That is identical to maximizing  $ROI_{BT}$ .
- But,  $NPV_{proj}$  represents potential gain to company if project is initiated. It should be maximized.

- One engineering compromise:

$$\text{maximize } NPV_{proj} \quad \text{s.t. } NPV_{\%} > NPV_{\%/acceptable}$$

where  $NPV_{\%/acc}$  is the maximum allowed level of risk.

- One can now develop acceptable levels of risk according to nature of product (commodity, high-value chemical, bio/pharma, etc. or use any other criteria.



Q: Did it all really happen this way?

A: Only if you were there at the time.

- Predicting the future of computing—a piece of cake?
- The first personal computer—Bendix G-15.
- Early mainframe stuff—IBM's and Univac's.
- God's gift to frustration—Analog & hybrid computers.
- Origins of the Internet—Arpanet and UCSB.
- Laboratory and real-time computers—ADCs/DACs, machine language, assembly language, #%&@!!
- Publish and perish. Too much of a good thing?
- Spread sheeting & plant profitability estimates.
- “What goes around, comes around:” Too much computing? => Re-focus on fundamentals.

# Highlights of a Long, Interesting Career

- First chem. eng'g. application of adaptive control: Adaptive control of pH system – 1964.
- Modeling of polycondensation reaction system using reduced-order (end group) relations: Ault & Mellichamp – 1972.
- Real-Time Computing Laboratory (1971-88), CACHE RT Monographs (1977-79) & Professional Textbook (1983) — lots of fun but dead-end professionally (except for PD&C textbook).

# Highlights of a Long, Interesting Career

## (cont)

- First Ch. E. experimental application of control to *unstable* distributed parameter system: Autothermal reactor with Bonvin, Wong, and Rinker (1980).
- *Process Dynamics and Control*, with Seborg and Edgar (1<sup>st</sup> ed., 1989; 2<sup>nd</sup> ed., 2003) with Seborg, Edgar, and Doyle (3<sup>rd</sup> ed., 2010); translated Japanese, Korean, and Chinese editions. (Particularly proud of two plantwide control chapters, now relegated to appendices in 3<sup>rd</sup> ed.)

# Highlights of a Long, Interesting Career

## (cont)

- Profitability research in conjunction with Mike Doherty/Senior Ch.E. Plant Design course. Research paper now in preparation links *profitability* and *investment risk* in the context of a *simple but exact linear model*.
- Modeling of reactor/separator/recycle process (with Ward, Griffin, Doherty) that shows all optimization of reactor alone is incorrect.

# Highlights of a Long, Interesting Career

## (cont)

- Biggest missed opportunity: In 1972 I published two articles on model predictive control. But the implementation was analog; the model used was a transfer function; the objective function was “minimum time.” With just three major changes, DMC => “Duncan Mellichamp Control.”
- Everything above ignores both my “university administrative career” and my ongoing “non-profit organization retirement career” (currently board president of Opera Santa Barbara).

# Finale

“What goes around, comes around:”

In final analysis, avoid computing “without thinking.”

Must conclude, my most useful analytical work utilized collapsed models, reduced-order models, even approximate models combined with analytical solutions, not brute force computing approaches.

So, perhaps good fundamental engineering analysis wins out, ... but working with computers for over 50 years has been both productive and a lot of fun.





Die Lustige Witwe – The Merry Widow – Opera Santa Barbara 2009