computers and people

Jan.—Feb., 1985

Vol. 34, Nos. 1-2

including *Computer Graphics and Art*

formerly *Computers and Automation*

KOMSOMOLSKAYA SUBWAY STATION IN MOSCOW'S METRO SYSTEM

Computer System Reliability and the Hazard of Nuclear War – *Prof. Alan Borning*

Bar Codes: Basic Principles – *Bruce R. Wray*

Computerization in the Moscow Subway – *Alexander Dedul*

Computers and the Japanese National Railway – *L. Fletcher Prouty*

Dangerous Fantasies – *Edmund C. Berkeley*

The Computer Almanac and the Computer Book of Lists – *Neil Macdonald*

The Computer Almanac and Computer Book of Lists – Instalment 39

Neil Macdonald
Assistant Editor

9 CHAIRMEN OF THE BOARD OF THE INFORMATION INDUSTRY ASSOCIATION (List 850101)

Starting	Chairman
1969	William T. Knox
1971	Jeffrey Norton
1973	Dr. Eugene Garfield
1975	Harold Redding
1977	Dr. Herbert R. Brinberg
1979	Robert F. Asleson
1981	Thomas A. Grogan
1983	Roy K. Campbell
1984	Norman M. Wellen

(Source: Information Industry Association, 316 Penn. Ave. S.E., #400, Washington, DC 20003)

11 REMARKABLE MEN RELATED TO COMPUTER DEVELOPMENT (List 850102)

John Napier (1550-1617) / slide rule and logarithms for analog multiplication, and "Napier's bones" for digital multiplication

Blaise Pascal (1632-1662) / numerical adding machine using gears with ten teeth and a carry tooth

G. W. von Leibniz (1646-1716) / multiplying and dividing machine for numbers using repeated addition and subtraction

Joseph Jacquard (1752-1834) / machinery for weaving intricate designs in tapestries on looms, using punched cards with holes for selection of needles bearing different threads

Charles Babbage (1792-1871) / a "Difference Engine", and an "Analytical Engine", partially constructed; perfectly conceived general purpose computer for numbers and numerical computations

Herman Hollerith (1860-1929) / designed and produced machinery for classifying, sorting, counting, and adding cards with punched holes representing instances and numbers; first used in the tabulation of the U.S. Census 1890; holder of nearly 50 patents on uses of punched cards for data processing; founded forerunner company to IBM

Norbert Wiener (1894-1964) / instructor at Mass. Inst. of Technology 1919-1961; published "Cybernetics", 1948, and "The Human Use of Human Beings", 1950; active in mathematics, philosophy of mechanistic and mathematical systems, feedback, automata, simulation of human thought processes; more than 100 publications

Howard H. Aiken (1900-1973) / head of the Harvard Computation Laboratory, Cambridge, Mass., 1939-61; here one of the first automatic digital computers, designed and engineered by Aiken and staff, called the Mark I, started operating in 1944; it was constructed with U.S. Navy and IBM (T. J. Watson) support

Wallace J. Eckert (1902-1971) / used punch card machines 1929-33 for interpolation of astronomical data, reduction of observational data, and numerical solution of planetary equations; published "Punched Card Methods in Scientific Calculation", 1940; director of U.S. Nautical Alamanac Office, 1940-45; head of IBM's Pure Science Department, and the T. J. Watson Scientific Computing Laboratory, 1946-67

John von Neumann (1903-1957) / active in logic, quantum theory, theory of high-speed computing machines, theory of games and strategy, applied mathematics; professor at the School of Mathematics, Inst. for Advanced Study, Princeton Univ.; planner of the Inst. for Advanced Study computer; author of over 500 papers

Alan M. Turing (1912-1954) / active in mathematics, computing machines, chess, cryptanalysis, code deciphering; located at the British National Physical Laboratory and elsewhere; inventor of the "Turing Machine", which expresses the mathematical notion of effective computability; originator of remarkable software

(Source: Neil Macdonald's notes)

22 OF THE OVER 100 PRESENTATIONS AT THE OCTOBER 1984 CONFERENCE OF THE COMPUTER AND AUTOMATED SYSTEMS ASSOCIATION OF THE SOCIETY OF MANUFACTURING ENGINEERS (List 850103)

Applications of Artificial Intelligence in Capital Intensive Process Industries / Dr. John P. Elwood

An Application of Expert Systems in Flexible Manufacturing Systems / Malcolm D. Hall

Executive Strategies for Computer-Integrated Manufacturing / Daniel P. Mincavage

Computer Integrated Flexible Manufacturing / George Hess

Flexible Manufacturing Simulation / John Bernard

Robots and Automated Systems for World Class Quality / Jack H. King

Control Software for an Advanced Sensor-Based Robotic Assembly Station / Karen A. Hope

Direct Computer Control / George W. Jones

Flexible Manufacturing Controller: The Ultimate Tool for Factory Integration / Meir Weinstein

A Conceptual Schema for a Computer Integrated Manufacturing Database / Francois B. Vernadat

New Generation Computers and their Implications for Computer Integrated Manufacturing / Daniel S. Appleton

Utilization of Artificial Intelligence in Manufacturing / David Liu

Real Time Simulation Eliminates the Risk of Industrial Automation / Max W. Hitchens

Integral Link Between Geometric Modeling and Computer Assisted Manufacturing Applications / A. Kader Elgabry

Integration of a Multi-Robot Process Line under Minicomputer Control / John F. Follin

Hard Automation vs. Robots -- Making the Choice / John W. Schott

Templates for an Integrated Common Database / B. Neil Snodgrass

Management Attitude to Flexible Manufacturing Systems / Venkitaswamy Raju

Blending Advanced Manufacturing Technologies with New Management Practice / Glen A. Allmendinger

Integrating Computer Simulation into a Small Manufacturing Operation / Robert M. Cowdrick, Jr.

Computer-Aided Production Engineering, and the Integration of Computer-Aided Process Planning, Engineering, and Manufacturing / Gayle L. Berry

Designing Flexible Manufacturing Systems Using Simulation / David B. Wortman

(Source: announcement of conference in Anaheim, CA, Oct. 1 to 4, 1984, of Computer and Associated Automated Systems Association of SME, One SME Drive, P.O. Box 930, Dearborn, MI 48121)

10 APHORISMS (List 850104)

When experience teaches, the test comes first, the lesson second.

The school of hard knocks teaches with impact.

If you can't say something in four sentences or less, save your breath to cool your soup.

Honesty is the best policy, except when you are telling white lies.

No excuse is better than any excuse.

If you could have half your wishes, you would double your troubles.

Home is where, when they feed you cauliflower, you have to eat it.

The postman brings, and the trashman takes away.

Almost nobody understands the first time.

It is remarkable how long misleading beliefs survive.

(Source: Neil Macdonald's notes)

Ω

Vol. 34, Nos. 1-2
January-February, 1985

"Computers and People" (ISSN 0361-1442), formerly "Computers and Automation," is published every two months at 815 Washington St., Newtonville, MA 02160, U.S.A., by Berkeley Enterprises, Inc. Printed in U.S.A. Second-class postage paid at Boston, MA and additional mailing points.

Subscription rates, effective Sept. 1, 1984: U.S.A., $18.50 for 1 year, $36.00 for 2 years; elsewhere, add $8.00 per year.

NOTE: The above rates do not include our publication, the "Computer Directory and Buyers' Guide." To receive this, please add $20.00 per year to your subscription rate in the U.S.A., and $23.00 per year elsewhere.

NOTE: No organization in Switzerland or Morocco is authorized or permitted by us to solicit for, or receive payment for, "Computers and People" or the "Computer Directory and Buyers' Guide." All subscriptions to and payment for these publications should be sent directly to Berkeley Enterprises, Inc.

Please address mail to: Berkeley Enterprises, Inc., 815 Washington St., Newtonville, MA 02160, U.S.A.

Postmaster: Please send all forms 3579 to Berkeley Enterprises, Inc., 815 Washington St., Newtonville, MA 02160, U.S.A.

Change of address: If your address changes, please send us both your new address and your old address (as it appears on the magazine address imprint), and allow four weeks for the change to be made.

computers and people formerly *Computers and Automation*
including *Computer Graphics and Art*

Computers and the Arms Race

7 Computer System Reliability and the Hazard of Nuclear War [A]
by Prof. Alan Borning, Univ. of Washington, Seattle, WA
Solutions to the problems of the arms race are sought in more accurate missiles and more complex computer hardware and software. But advanced technology offers only the illusion of a solution; there are many instances of computers giving inaccurate information and yielding disastrous or near-disastrous results. Solutions to political problems must be sought in the political realm, not the technological.

6 Dangerous Fantasies [E]
by Edmund C. Berkeley, Editor
Large computer systems may often give correct answers, but they aren't infallible. Because they are very complex, the chances of error are greatly increased. And the impossibility of testing certain large systems, like those for attacking many missiles, makes them even more undependable. Indeed, to rely on the perfection of such a system is to invite disaster.

Computers and Transportation

1,5,16 Computerization in the Moscow Subway [A]
by Alexander Dedul, c/o USSR Embassy, Ottawa, Ont., Canada
The Metro subway system in Moscow provides its passengers with fast, reliable and affordable transportation. With computers monitoring train speed and spacing and modern coaches providing faster acceleration and passenger comfort, Muscovites save an estimated total of 800 million hours per year by riding the Metro.

20 Computers and the Japanese National Railway [A]
by L. Fletcher Prouty, Alexandria, VA
The Japanese began planning their modern rail passenger service before World War II. Today they provide fast and safe passenger service by using proper track, automated computerized maintenance, train monitoring and scheduling. The Japanese system sets an example for efficiency and profitability in rail service for the world.

Computers and Productivity

12 Bar Codes: Basic Principles [A]
by Bruce R. Wray, Computype, Inc., St. Paul, MN
Bar coding is a fast-growing technology which is finding its way into grocery stores, libraries, government agencies, and elsewhere. Because data is collected and processed quickly, accurately and easily, the use of bar coding saves money and greatly increases productivity, sometimes as much as 400%.

The magazine of the design, applications, and implications of information processing systems – and the pursuit of truth in input, output, and processing, for the benefit of people.

Computer Applications

24 **Computer-Simulated Robots** [N]
based on a report in the *Financial Times*, London, England

24 **Network of 400 Personal Computers with LISP Language** [N]
by Robert M. Byers, Mass. Inst. of Technology, Cambridge, MA

26 **Computer-Controlled Robot to Sense the Environment Being Designed at Univ. of Pennsylvania** [N]
by Virgil Renzulli, Univ. of Pennsylvania, Philadelphia, PA

Lists Related to Information Processing

2 **Computer Almanac and the Computer Book of Lists – Instalment 39** [C]
by Neil Macdonald, Assistant Editor
9 Chairmen of the Board of the Information Industry Association / List 850101
11 Remarkable Men Related to Computer Development / List 850102
22 of the Over 100 Presentations at the October 1984 Conference of the Computer and Automated Systems Association of the Society of Manufacturing Engineers / List 850103
10 Aphorisms / List 850104

Computers, Games and Puzzles

28 **Games and Puzzles for Nimble Minds – and Computers** [C]
by Neil Macdonald, Assistant Editor
MAXIMDIDGE – Guessing a maxim expressed in digits or equivalent symbols.
NAYMANDIDGE – Discovering a systematic pattern among random digits.
NUMBLE – Deciphering unknown digits from arithmetical relations among them.

Announcement

The Computer Directory and Buyers' Guide

The names, addresses and descriptions of over 3600 computer field organizations have been inserted and updated into our computer data base for the next Directory edition. Production of the photooffset master for printing, however, has been delayed. We hope that we will have this, the 27th edition, ready for mailing to subscribers early in 1985. This edition of the *Computer Directory and Buyers' Guide* will now be entitled "1984-85".

Meanwhile, any current subscriber to *Computers and People with Directory* who does not already have a copy of the 1983 *Computer Directory and Buyers' Guide* may on request to us receive a copy of that issue, so long as the overrun lasts.

Front Cover Picture

The front cover picture shows the inside of the Komsomolskaya subway station in Moscow's Metro system. This station is a work of art. Its lighted chandeliers, the ceiling's molded stucco and the marble walks along both sides of the station make it one of the most attractive stations in the entire computerized, 200-kilometer network of 123 stations. For more information, see page 16.

Editorial Note: We invite articles on the subject of computers and nuclear weapons. Computers, and computer people who work to make nuclear weapons work, are an essential ingredient of the nuclear evil.

There will be zero computer field and zero people if the nuclear holocaust and the nuclear winter occur. Every city in the United States and in the Soviet Union is a multiply computerized target. Thought, discussion, and action to prevent this holocaust is an ethical imperative.

Key

[A]	–	**Article**
[C]	–	**Monthly Column**
[E]	–	**Editorial**
[EN]	–	**Editorial Note**
[F]	–	**Forum**
[FC]	–	**Front Cover**
[N]	–	**Newsletter**
[R]	–	**Reference**

Notice

*D ON YOUR ADDRESS IMPRINT MEANS THAT YOUR SUBSCRIPTION INCLUDES THE COMPUTER DIRECTORY. *N MEANS THAT YOUR PRESENT SUBSCRIPTION DOES NOT INCLUDE THE COMPUTER DIRECTORY.

Editorial

Dangerous Fantasies

Edmund C. Berkeley, Editor

Persons who come into contact with large computer systems often hope and trust that the computer systems will give correct answers invariably. But this is a fantasy, and can be dangerous. It is true that large computer systems very often have a remarkably high proportion of correct answers. But there is a long record of mistakes. The promise of computers is wonderful, but the reality is less than wonderful.

Why?

There are a number of reasons.

Complexity. The complexity of a computer program is often measured in the number of lines of instructions, all of which have to be correct and all of which have to be in the right sequence. When the number of lines of instructions for the operation of a computer system goes over 10,000 the imagination and the capacity of the team of programmers is strained; when it goes over 100,000, the intricacy is almost overwhelming; and when it goes over a million, success in design is very likely to be out of reach.

Bias. A government department may have a contracting supervisor who feels committed to a philosophy of "can do": "the difficult we do at once; the impossible takes a little longer." Praiseworthy though this attitude may be, it leads to bias, the appraisal of problems and their solutions on a less than objective basis. Under such conditions, there is a fair chance that a computer system being constructed will not meet all its requirements.

Testing. The regular first test of an elaborate computing system is to take a sample case and ask "Does it work?" And if it works on the first sample case, does it work on many other sample cases? But there are computer systems (and other systems as well) which cannot be tested. For example, suppose there were 100 ballistic missiles being launched during 10 minutes by a foreign power, and the problem was to seek them all out by means of a computer system and disable them. No proposed solution could possibly be tested.

Probabilities. Another reason why intricate computer systems cannot always be reliable is often expressed in Murphy's Law, "If something can go wrong, it will." Another way of expressing this principle is "If there is a 99.99 percent chance that each of 10,000 steps will go right then the chance of all 10,000 going right is only 1 in 3." Accuracy is a difficult undertaking and the principles of probability are in league against it.

So much for general argument. What about some examples?

The October 1960 Alarm. On October 5, 1960, the station of the Ballistic Missile Early Warning System at Thule, Greenland, picked up signals which were analyzed by the computers there as a flight of missiles coming up over the horizon from Russia and heading in the direction of America. Thule messaged this to the North American Air Defense Command (NORAD) in Colorado. In NORAD, the duty officer "refused to be panicked" and telephoned Thule, who shortly determined their error, that they had picked up a large earth satellite called the moon.

The June 1980 Alarm. On June 3, 1980, at 1:26 am, the display systems of the Strategic Air Command (SAC) in Nebraska indicated that two submarine-launched ballistic missiles were headed towards the United States. SAC personnel called the North American Aerospace Defense Command Center (NORAD) in Colorado. NORAD said they had no indication of missile launches. Shortly, SAC display systems again indicated that intercontinental ballistic missiles had been launched towards the United States. Then the display at the National Military Command Center (NMCC) in the Pentagon indicated that submarine-launched ballistic missiles were headed towards the United States. Then the top duty officers of NORAD, SAC, and NMCC

(please turn to page 15)

Computer System Reliability and the Hazard of Nuclear War

Prof. Alan Borning
Computer Professionals for Social Responsibility
Dept. of Computer Science FR 35
Univ. of Washington
Seattle, WA 98195

"People cannot make decisions of great scope and consequence in a matter of minutes. . . . Computers are no better at such decisions than the programmers who write their software."

Based on an article in *IPPNW Report* for Oct. 1984, published by International Physicians for the Prevention of Nuclear War, 225 Longwood Ave., Boston, MA 02115, and reprinted with permisssion. For references, see *IPPNW Report*, page 21.

False Alerts

On several occasions, the NORAD early warning system has mistakenly indicated that Soviet missiles were headed for the United States. These incidents raise questions of the following sorts: Could a computer failure, in either the American or the Soviet warning systems, start an accidental nuclear war? What risks are associated with placing the nuclear forces of one or both powers on alert? Would it be responsible for a country to adopt a policy of launch-on-warning, in which missiles would be fired based on warnings that an attack was imminent?

Computers are used extensively in military applications: for guiding missiles, analyzing sensor data and warning of possible attack, controlling communications systems, managing data on friendly and enemy forces, simulating possible battles, as well as for such mundane tasks as keeping track of personnel, inventories, and payrolls.

In looking at the military uses of computers, I see a number of disturbing facts and trends. As a practitioner in the field, I want to encourage examination of questions relating to computing and the threat of nuclear war, both within the profession and among the general public. In this paper I examine only the nuclear forces of the US and the USSR. The warning systems and nuclear forces of other countries clearly add to the problems described here.

The June 3, 1980 Events

On Tuesday, June 3, 1980, at 1:26 am, four stories underground at the command post, Strategic Air Command, Offutt Air Force Base, near Omaha, Nebraska, the display system indicated that two submarine-launched ballistic missiles (SLBMs) were headed toward the United States. Eighteen seconds later, the system showed an increased number of SLBM launches. SAC personnel called the North American Aerospace Defense Command Center, located 1200 feet under the solid granite of Cheyenne Mountain in Colorado. NORAD stated that they had no indication of SLBM launches. After a brief period, the SAC screens cleared. Shortly, the warning display at SAC indicated that Soviet intercontinental ballistic missiles had been launched toward the United States. Then the display at the National Military Command Center in the Pentagon indicated that SLBMs had been launched. The SAC duty controller directed all alert crews to move to their B-52 bombers and to start their engines, so that the planes could take off quickly and not get blasted on the ground by a nuclear weapon. Land-based missiles were brought to a higher state of readiness, and battle-control aircraft prepared for flight. In Hawaii, the airborne command post of the Pacific Command took off, ready to pass messages to US warships if necessary.

While all this was happening, a Threat Assessment Conference was convened among the top duty officers at NORAD, SAC, and the NMCC. For the next three minutes, there was discussion among the three officers. There were a number of factors that made them doubt that an actual attack was under way: NORAD itself had no indications of an attack, the indications on the displays at

SAC and NMCC did not follow any logical pattern, and the different command posts were receiving different information. Three minutes and twelve seconds into the alert, it was cancelled. It was a false alert.

NORAD left the system in the same configuration in hopes that the error would repeat itself. The mistake was reproduced three days later, on June 6 at 3:38 pm, with SAC again receiving indications of ICBM attack. Again, SAC crews were sent to their aircraft and ordered to start their engines.

The cause of the incidents was eventually traced to the failure of a 74175 integrated circuit chip in a Data General computer used as a communications multiplexer. This machine took the results of analysis of sensor data and was part of the system that transmitted it from NORAD to SAC, NMCC and Canadian Headquarters in Ottawa. The communications links were tested by means of sending filler messages. At the time of the false alerts, these filler messages had the same form as attack messages, but with a zero filled in for the number of missiles detected. The system did not use any of the standard error correction or detection schemes for these messages. When the chip failed, the system started filling in the "missiles detected" field with random digits.

These false alerts received considerable press attention at the time. As a result of the publicity, on June 20 Senators Gary Hart and Barry Goldwater were asked to investigate the incidents by Senator John Stennis, Chairman of the Senate Committee on Armed Services. They prepared both classified and unclassified versions of a report; the unclassified report was the principal source of information for the above account of the incident.

The Nov. 9, 1979 Events

The incidents of June 3 and 6, 1980 illustrate one sort of error that can cause a false alert: a hardware failure. Another incident illustrates another sort of error: human error.

On November 9, 1979, a test tape containing simulated attack data, used to test the missile warning system, was fed into a NORAD computer, which, through human error, was connected to the operational missile alert system. During the course of the six-minute alert, ten tactical fighter aircraft were launched from bases in the northern United States and Canada, and, as in the June 1980 incidents, a Threat Assessment conference was convened.

What about similar failures in the Soviet warning systems? I have been unable to document whether or not such failures have occurred. (Hints of the U.S. warning system failures leaked to the press; the Pentagon stated that they would otherwise not have been made public. At a news conference shortly after the June 1980 incident, Assistant Secretary of Defense Thomas Ross would not say whether the US knew about similar false alerts in the USSR. The state of the art in Soviet computer science lags several years behind that in the US. However, the NORAD computers are very old by computing-industry standards (one Congressional Report termed them "dangerously obsolete"); whereas Soviet military computers are on the leading edge of their technology.

Unsettling as the false alerts in November 1979 and June 1980 were, in the opinion of most reviewers of the incident, including myself, the United States was nowhere near to launching its missiles and starting World War III. Most importantly, human judgment played an essential role in the procedures followed in the event of an alert, and these procedures provided enough time for the humans involved to notice that a computer system was operating incorrectly. Also, NORAD procedures called for confirmation of the attack by an independent system, e.g., radar systems that would observe the attacking missiles in flight, and the chance of simultaneous false alerts for both systems under normal circumstances is very small.

The Nov. 5, 1956 Events

A further danger comes from the possibility of compound stimuli to the system, perhaps from ambiguous or incomplete intelligence information. One such example occurred in 1956, at the time of the Suez Crisis and Hungarian uprising. On the night of November 5, the following four coincidental events occurred. First, US military command headquarters in Europe received an urgent message that unidentified jet aircraft were flying over Turkey. Second, there were additional reports of 100 Soviet MiG-15 fighters over Syria. Third, there was a report that a British bomber had been shot down over Syria (presumably by the MiGs). Fourth, there were reports that a Russian naval fleet was moving through the Dardanelles, perhaps to leave the Black Sea in preparation for hostilities. General Andrew

Goodpaster was reportedly afraid that the events "might trigger off the NATO operations plan," which at the time called for a single massive nuclear attack on the Soviet Union.

As it turned out, all four reports were incorrect or misinterpretations of more innocent activities: the jets over Turkey turned out to be a flock of swans; the MiG's over Syria were part of an official escort for the Syrian president; the British bomber was downed by mechanical difficulties; and the Russian fleet was on a scheduled exercise. In Bracken's words, "the detection and misinterpretation of these events, against the context of world tensions from Hungary and Suez, was the first major example of how the size and complexity of worldwide electronic warning systems could, at certain critical times, create a crisis momentum of its own."

The worldwide electronic warning and communications systems of today are immensely more complex and reactive than those of 1956. In conjunction with fears of a first strike and the necessarily short reaction times, there is much ground for concern. Events that are in actuality unrelated may seem to be part of a larger pattern. Once the nuclear forces are placed on alert, further human or mechanical errors may occur. After the June 1980 incident, the Hart-Goldwater report notes that "Even though the command post controller prevented any undue reaction to the false and erroneous data, there seemed to be an air of confusion following the determination that the data were erroneous." It is likely that the "air of confusion" would be much worse if it were suspected that the indications of attack might be real.

To be at all confident of the reliability of complex systems, there must be a period of testing under conditions of actual use. As far as is publicly known, the command and control systems of the US and the USSR have never been "tested" under conditions of simultaneous high alert; in fact, the highest level of conference in the US missile warning system, the Missile Attack Conference, has never been called. Further, in a crisis situation, the very short times available for military personnel and national leaders to react and make decisions will undoubtedly lead to poorer judgment than under more normal circumstances, increasing the chances of misinterpretation of data and of error in operation of systems. (Psychologists have repeatedly noted that the quality of human judgment deteriorates under pressure of time, becoming much worse when only a few minutes are available to evaluate and react to a situation.)

The combination of the untestability of the warning and control systems under highly stressed conditions and the short times available for making decsisions is grounds for considerable concern.

Launch-on-Warning Strategy

Launch-on-warning is a strategy for retaliation for a nuclear attack. Under this strategy, retaliatory missiles are launched in response to sensor indication that enemy missiles are on the way, before the warheads on the attacking missiles have detonated. This strategy stands in contrast to "riding out the attack," a strategy in which a nation would absorb a full nuclear strike, and would retaliate only after positive verification had been obtained that an attack has taken place.

An obvious disadvantage of launch-on-warning is the possibility of a retaliatory strike triggered not by an enemy attack but by computer or other error. Why, then, would one consider adopting such a strategy?

The land-based missiles of both the United States and the Soviet Union have been growing more accurate over the years. For example, the US Minuteman III MK12 missile has an accuracy of 280 meters, the older Soviet SS-11 Mod 1 an accuracy of 1400 meters. The Pershing II missile is even more accurate. It uses a new guidance technology in which live radar images of the landscape surrounding the target area are compared with internally stored map information during its descent, so that course corrections can be made before impact. Its accuracy is reportedly 30-40 meters.

This increased missile accuracy puts at risk all fixed targets, such as land-based missile silos and command centers, even highly hardened ones. While it is not at all certain that this vulnerability of fixed targets implies that a first strike could be successfully launched, nevertheless strategic planners in both the US and the USSR have been concerned for decades with the problem. One way of dealing with the problem is launch-on-warning. If one side believes that an enemy attack is coming, retaliatory missiles can be launched and on their way, leaving the attacking warheads to explode on empty missile silos.

Although weapons based on submarines at sea and on aircraft are not currently so threatened, the present US doctrine calls for all three "legs of the strategic triad" to be capable of inflicting retaliation. The risks to deterrence are more acute for the Soviet Union, which has a higher proportion of its strategic nuclear weapons on land-based missiles.

Adoption of a launch-on-warning policy would be a dangerous action, because of the danger that a false alert from a missile warning system would trigger a retaliatory response. If launch-on-warning were adopted, it almost certainly would be activated only in times of crisis, rather than continuously, to reduce the risk of accidental war. (Note that a policy of activation on this basis is an admission of distrust in the complete reliability of the warning systems!) Nevertheless, in my opinion, it is simply not an acceptable policy.

While the strategy is certainly possible in theory, because of the very short times involved there are doubts that launch-on-warning is a practical policy, at least if an acceptable level of control is to be maintained on the nuclear forces of the country that adopts it. From a broader viewpoint, launch-on-warning can be seen as one extreme at the end of a spectrum of policies for retaliation, the dimension of the spectrum being how long a power waits to respond when it believes that an attack is imminent or under way. Taking this broader view, pressures toward launch-on-warning are a symptom of underlying problems: strategic doctrine that holds that military assets at known, fixed locations (land-based ICBMs and command posts) are an essential part of a nation's nuclear forces; the perception that the vulnerability of fixed targets is a pressing problem; new weapons systems that make them more vulnerable; and the consequent decrease in time available to make decisions in nuclear crises.

Would it be responsible for either the USSR or the US to adopt weapons systems and policies that rely on computer systems, such as missile warning systems, functioning without failure? In this section I will argue that it is not. I will not attempt to show that it is certain that failures will occur in complex military systems, but rather that there is room for doubt that adequate reliability can be achieved. The standard of reliability required of a military system that has the potential of triggering a thermonuclear war if it fails must be higher than that of any other computer system, since the magnitude of disaster is so great.

Power Blackouts and Three Mile Island

Much research and development effort has been devoted to the construction of reliable computer systems, and some impressive results have been achieved. However, there have also been some impressive failures of computer systems designed to be reliable: the NORAD false alerts described earlier, the total collapse of a US computer communications network (the ARPANET) in October 1980 due to a gridlock-like phenomenon, and problems with backup computer synchronization that, at the last minute, delayed the launch of the first Space Shuttle.

Outside of the realm of computer systems, two instructive accidents with widespread consequences are the power blackout in the Northeastern United States and Ontario, Canada in 1965, and the accident at the Three Mile Island nuclear power plant in 1979. In each of these cases the seriousness of failure was well understood in advance and many precautions had been taken in the system design.

The 1965 Northeast power blackout started when a backup protective relay on a 230-kilovolt transmission line at the Beck Hydroelectric Plant in Ontario operated when the current flowing through the line exceeded the relay's setting. This disconnected the transmission lines then moving power north from that plant, reversing the power flow from north to south and causing a massive surge of power into the Northeastern United States. The disruptions quickly spread to encompass an area of some 80,000 square miles, directly affecting an estimated 30 million people in the United States and Canada. After the event, it was claimed that system modifications made it impossible for a similar accident to reoccur; but one did, in July 1977 in New York City. On that occasion, a succession of lightning strikes that "just never happens" (in the words of the president of Consolidated Edison Company) knocked out parts of the system; in trying to handle the outages a series of overloads occurred that eventually brought down the New York area power system.

The Three Mile Island accident began with an equipment failure (of a relief valve), but its severity was much compounded by subsequent operator error.

In hindsight, blame can be assigned to individual component failures or specific human errors in each of the above incidents. But that is almost always the case with design errors. In designing automatic systems we must anticipate all possible eventualities and specify what should happen in all cases. The real culprit is simply the complexity of the systems, and our inability to think through in advance and plan for all of the things that can go wrong.

Sources of System Failure

The sources of computer system failure include hardware failure, hardware design errors, software design errors, and human error (e.g. incorrect operation or maintenance).

Hardware failures are perhaps the most obvious cause of system failures, as in the NORAD failures of June 1980. Individual components can be made very reliable by strict quality control and testing, but in a large system it is not reasonable to expect that no component will ever fail. Other techniques for dealing with local failures -- replication, weighted voting, codes for error detection and correction, dynamic reconfiguration, and so forth -- are all valuable, and have been used in the construction of very reliable devices. However, when one builds very complex systems -- and a missile warning system in its entirety is certainly an example of a complex system -- one becomes less certain that one has anticipated all the possible failure modes, that all the assumptions about independence are correct.

Another potential cause of failure is a hardware design error. Again, the main source of problems is not the operation of the system under the usual, expected set of events but its operation when unexpected events occur. For example, timing problems due to an unfortunate set of asynchronous parallel events that occur very seldom are particularly hard to find.

It is in the nature of computer systems that much of the "design" is embodied in the computer's software. Because that software is relatively easy to modify we can change the system design quite readily. This means that we can correct software design errors easily, but it also means that we can just as easily introduce design errors. Because most of the complexity of a computer system is usually contained in its software, that is the part that most often breaks when some unanticipated circumstance arises. Anyone who has worked on a large computer system knows how difficult it is to manage the development process. Usually, there is nobody who understands the entire system completely. A variety of strategies are currently used to help cope with these problems: high-level languages, modular design, information hiding, and so forth. Nevertheless, it is widely acknowledged that the whole process is not really satisfactory.

Other techniques, such as proofs-of-correctness or automatic programming, may help in the long term. (In a proof-of-correctness, either a human or a computer proves mathematically that a program meets a formal specification of what it should do; in automatic programming, the program is written automatically from the specification.) However, these techniques are still very much in the research stage. For example, simple compilers have been proven correct, but programs of the complexity of the real-time satellite data analysis programs are well beyond the state of the art. Automatic programming is even less advanced.

How Does One Know the Specification Describes What One Wants?

However, there is a more fundamental problem with techniques such as proofs-of-correctness and automatic programming. A proof-of-correctness, for example, shows that one formal description (the specification) is equivalent to another formal description (the program). How does one know that the specification describes what one wants? Are there events that may occur that were simply not anticipated when the specification was written? For example, in 1960 a false alert in the BMEWS radar system in Greenland was triggered by the rising of the moon; another false alert in the '50s was caused by a flock of geese. Proving that the system met its spefifications would not help if nobody thought about the rising moon or geese when writing the specification.

Yet another source of failure is human error, as in the November 1979 false alert. People do make mistakes, despite elaborate training and precautions. In time of stress and crisis, such mistakes become more likely. There are some worrying statistics about alcohol, drug abuse, and aberrant behavior among military personnel with access to nuclear weapons. Alcoholism is a major health problem in the Soviet Union, and is at least as likely to exist in the Soviet military as in that of the US.

(please turn to page 27)

Bar Codes: Basic Principles

Bruce R. Wray, Dir. of Marketing
Computype, Inc.
2285 W. County Road "C"
St. Paul, MN 55113

"The technology for bar code scanning systems is straightforward and available now. . . . and the speed of data capture is 3 times faster than a skilled keypunch operator."

Based on an article "Bar Code Basics" by Bruce R. Wray in *Recognition Technologies TODAY* for October, 1984, published by and copyright by Recognition Technologies Users Association, P.O. Box 2016, Manchester Center, VT 05255, and reprinted with permission.

I want to deal with four topics: Why bar codes? How do bar codes work? Where are bar codes used? And economic justifications.

Why Bar Codes?

Bar coding is all about automated data collection. It's a method for rapidly, accurately, and efficiently gathering data from the environment, and putting it into some permanent form for subsequent processing by computer. It is most certainly not the only method currently available to automatically capture data. Let's take a look at two other possible methods.

Keyboard entry and manual methods. These are perhaps the most familiar methods of capturing data. Operators sit at keyboards and enter strings of letters and numbers that identify specific products or transactions. One character per second to several characters per second is the speed range commonly associated with manual entry. A good rule of thumb is two keystrokes per second. Another good rule of thumb is one error per 100-300 keystrokes. Let's face it -- it's a manual process and people make mistakes.

Magnetic recording. This method is most commonly used on credit cards to identify the account number of the cardholder. Several problems with MICR; the message must always be in exactly the same place; it's very expensive to produce; it's not particularly secure; and contact must be made with the paper in order to enter the information.

Over the past several years, bar codes have become the fastest growing method of automated data collection. They provide fast, accurate, efficient data collection, using technology that is reliable and equipment that is easily used with a minimum of training.

How Do Bar Codes Work?

The basic principle of bar coding is simple -- light is reflected in different amounts by different colored surfaces. A bar code scanner, either a hand-held wand, a laser, or a fixed- or moving-beam scanner, merely translates reflective differences into electrical signals.

Let's take a closer look at how it works. Our example will be based on the use of a hand-held bar code wand; the principles are the same no matter what type of scanner is used.

A small spot of light, which is passed over a series of dark bars and intervening white spaces, will reflect back into the scanner varying amounts of light -- lots of light will be reflected from the white spaces, but very little will be reflected back from the dark bars. (This is because dark colors tend to absorb more light than they reflect, while the opposite is true for light colors: they tend to reflect more light than they absorb. That, in fact, is what makes light colors light and dark colors dark -- how much light each reflects.)

These differences in reflected light are translated into electrical signals by the light detector, a photo diode. How does this work? Well, unlike a human eye, the scanner does not recognize the vertical edges of dark bars. As a scanner is moved from a light space onto a dark bar, the detected light decreases gradually. When this decrease is detected and light reaches a predetermined level, a logic decision is made to recognize the dark bar. Further movement of the scanner onto the dark bar continues to cause a further decrease in detected light. As the scanner continues to move, it encounters the opposite edge of the bar and begins to see the light space. The detected light begins to increase. Again, when it reaches a predetermined value, a logic decision is made to recognize the end of the dark bar and the start of the light space.

Conversion of Signal

The signal output from the wand detector is an analog signal and must be converted to a digital signal for interpretation and processing by the data terminal. This analog to digital conversion is performed by the electronics contained in the wand itself. (Some bar code wands are digital and do not need the analog to digital conversion.) The digital signals, for the purposes of most bar code scanning, represent binary 1's and 0's. Combinations of 1's and 0's are then translated into specific numbers and letters according to the particular rules of the bar code "language" or symbology being used.

The width of the narrowest element in a bar code is called the "X dimension." It is the X dimension which determines the density of the bar code, or how much information can be packed into a given amount of space. The narrower the X dimension, the higher the density of the bar code, and therefore the more information can be packed into, say, a linear inch. It's very similar to handwriting -- when you are writing, the smaller you make each letter, the more words you can get into a given space. High-density bar codes usually can encode about 10 characters of information into one inch of space.

Voids and Specks

Let's look at some of the possible problems that can be encountered when scanning a bar code symbol. Voids and specks -- these terms refer to the absence of ink where there should be ink, and the presence of ink where there shouldn't be ink. These conditions, or general edge roughness, can cause serious scanning difficulties. As the spot of light is being passed through the bar code symbol, suppose it encounters a speck of ink in the middle of what was supposed to be a wide space. The wand can easily be fooled into decoding that speck as a narrow bar. Similarly, if the spot of light is traversing a wide bar and encounters a void of sufficient size, all of a sudden what was supposed to be read as a wide bar is mistakenly read as two narrow bars with an intervening space.

The people who have written the most popular bar code symbologies have attempted to solve this problem by incorporating what is called a "self-checking" feature into the bar code. This means that each portion of the bar code representing a single character can stand alone -- in other words, there are sufficient checks to ensure that a single character can both be recognized and identified. Let's take a quick example from a bar code that is self-checking called "2 of 5." In 2 of 5, each character is made up of 5 bars, 2 of which must be wide (hence the name 2 of 5). Figure 1 shows the character 6 with a printing void in one of the wide bars. A scan line passing through this void would see a character composed of 4 narrow bars and 1 wide bar. Since all valid characters require 2 wide bars, this would result in what we call a non-read. This is the second worst thing that can happen when a bar code symbol is scanned. Now let's look at the worst thing that can happen: two independent printing defects occurring along the same scan line within the same character could produce a character substitution error (see Figure 2). This would result in what we call a mis-read -- and that is the worst that can happen, because you've entered incorrect information into the system.

Character "6" With a Printing Void

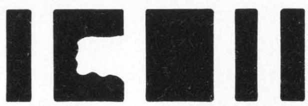


Figure 1

Character "6" Transposed into a "4" By Two Printing Defects

Figure 2

Image Quality

This is a good time to talk about the whole issue of bar code image quality. What do we mean by quality? Several factors play a part: dimensional accuracy and consistency -- the actual width of each element in the bar code must be within specific tolerances or the bar code will not scan. This accuracy must be characteristic throughout the bar code symbol, and from symbol to symbol.

The second factor is just overall clarity. Specks, voids, edge roughness, image spread or shrink, show-through, and so on, are all examples of poor printing quality which can seriously reduce both accuracy and efficiency within the system. The third factor affecting bar code image quality is called the "print contrast ratio." As you will remember from the early parts of this article, the basic principle of bar coding is light reflecting back from different colored surfaces in different amounts. This difference in reflectivity between the dark bars and the light spaces is expressed mathematically by the print contrast ratio. A spot of light is projected on a light surface, and the amount of light reflected back is measured. The same spot of light is then projected onto a dark bar, and the light reflected back is measured. The calculation is done, and the result is expressed as a percent -- the print contrast ratio. The people who manufacture bar code scanning equipment usually say a PCR of about 75% is minimum -- but that the higher the print contrast ratio, the closer to 100% first-read rates can be expected, and the more accurate the scanner will be in decoding the bar code symbol.

Where Are Bar Codes Used?

Now that we have a good idea of how bar codes work, we'll look at some of their common applications.

Probably the most familiar use of bar codes is the UPC of Universal Product Code used in retail grocery stores and supermarkets. The first five digits of the UPC symbol denote the manufacturer of the product; the remaining five digits represent that particular product, size, etc. When you shop for groceries at a store with scanning at the check-out counter, a laser scanner reads the bar code symbol on your package of Cheerios, does a look-up to determine current price, and also may automatically deduct one 12-ounce package of Cheerios from the store's inventory.

Libraries across the U.S. and Canada were quick to recognize the benefits of bar code scanning in controlling their collections of books and periodicals. Each volume in their collection is given a unique number, and the appropriate bar code label attached. Automated circulation systems provide real-time data to the library; and when you go to check out your books and they scan the bar code label on your library card, the system can automatically check its memory and determine if you have any overdue books. It's getting harder and harder to be dishonest these days!

Blood banks all over the world were also early in recognizing the benefits of bar code scanning. Each individual unit of blood is given a unique number, and blood products spun off from that whole blood are each given a product code. This facilitates blood inventory, cross-matching, and helps ensure that no transfusion errors are made.

In 1981, the Department of Defense completed a massive study of bar codes and possible applications in a number of their facilities. Now, if you are a supplier to the DoD, you are required to put a bar code label on each package you send to them; we'll talk about their anticipated cost savings from this project in our section on economic justifications.

A wide variety of manufacturing and distribution industries have also selected bar code scanning as an excellent way of improving accuracy, efficiency, and overall plant productivity. We are currently supplying bar code labels designed for use on automobile engines, printed circuit boards, warehouse totes, cattle, nuclear waste containers, filing cabinets and office furniture, sea-going shipping containers, and a host of others. It's now safe to say that there is probably no system or process that could not benefit from the addition of a bar code scanning system.

Economic Justifications

To paraphrase a famous statement of Mark Twain's, "Everybody talks about productivity but nobody does anything about it." It is certainly no secret that productivity growth, measured by output per employee-hour, has been declining in the U.S. Many reasons have been suggested for the decline, from tax structure disincentives to a "new breed" of worker in the labor force. Whatever the underlying causes, business people today are almost universally interested in increasing their own firm's productivity. Enlightened businesses the world over are making significant commitments to a proven method of productivity improvement -- bar code scanning systems for automatic identification.

Many products, programs, and schemes have been touted in the past as "The Answer" to the productivity problem. Most have failed the crucial test of time: they have worked for a while and then stopped working. Or, worse yet, their claims were overblown and the results only a fond wish. Bar code scanning systems, on the other hand, are proving themselves daily as effective tools for increasing productivity. Let me give you several examples. The supermarket industry has done the most analysis of productivity improvements because of the relatively long time they have been involved in bar code scanning. A recent study revealed the following benefits from scanning:

> Greater understanding of item movement and merchandising needs, which resulted in one brand being dropped because UPC data said more of the item movement stemmed from theft than sales;
>
> Checker productivity gains, measured in terms of actual check-out item counts per hour, of 17-30 percent;
>
> One retailer has been able to increase overall gross margin by 2% through scanning.

Another example in the retail area was cited in the "Wall Street Journal" (April 14, 1981). Giant Foods, Inc., a major user of bar codes and scanning check-out lanes, is cutting prices of 1,500 to 2,000 fast-moving grocery items because its labor costs are down. Giant says it is exploiting a distinct competitive advantage: All its grocery stores have bar code scanner check-outs -- many of its competitors' stores do not.

Productivity Improvement

As I mentioned earlier, the major confirmation of cost savings and productivity improvement through bar coding in the industrial area has been the major study done by the Department of Defense. Entitled "Logistic Applications of Automated Marking and Reading Symbols," (or LOGMARS for short), it outlined the specific areas within DoD that could utilize bar code scanning, and the economic benefits that could be expected. The following areas were used in the study: Receiving, Shipping, Inventory, Location Audit, Maintenance, Disposal, Sortation, Transportation. The basic approach of the cost-benefit study was to compare the current costs and methods to that of a proposed bar code system. Total estimated annual DoD tangible savings for use of bar code scanning instead of conventional methods of data entry were $113.9 million, over a 10-year period. In some functions, productivity was expected to increase by 400%.

Bar code scanning systems increase productivity in three relatively simple ways:

1. Speed. Automated scanning of items passing on a conveyor belt needs no human operator. Using a hand-held wand to read a bar code symbol on a small object typically takes two seconds per item. Assuming bar code symbols of 12 characters, the speed of data capture would be six characters per second, three times faster than a skilled keypunch operator.

2. Accuracy. Manual data entry tends to be inaccurate, with one error for every 100-300 keystrokes. With a high quality bar code image, error rates are approximately one error for every 300 million characters. With the addition of a check digit, which simply allows the scanner to mathematically determine that it has read the symbol correctly, the substitution error rate of Code 3/9. for example, is one error per billion characters!

   Some of our Codabar symbols were tested by a major scanner manufacturer; after 30 million scans, not one error in data collection had occurred!

3. Reliable, Easy-Use Technology. Bar code scanning systems are not based on technology that will be available tomorrow; the technology is straightforward and is available now. And you don't need to be a mechanical engineer to perform scanning -- six minutes is the average amount of time required to become proficient in hand-held bar code scanning.

Automated identification systems have a greater positive impact on productivity than perhaps any other single tool. Bar codes will continue to play a key role in productivity in a variety of applications for years to come. Ω

Editorial – *Continued from page 6*

conferred. They recognized that NORAD had no indications of an attack, that the different command posts were receiving different information, and the displays did not follow any logical pattern, and so after 3 minutes and 12 seconds the alert was cancelled.

In the article by Alan Borning in this issue, there are many more examples of failures of elaborate computer systems.

The conclusion we need to draw is that any elaborate computer system is likely to fail occasionally, and to rely on its perfection is to invite disaster. Ω

Computerization in the Moscow Subway

Alexander Dedul
Moscow
c/o USSR Embassy
400 Stewart St., Suite 1108
Ottawa, Ont. K1N 6L2, Canada

"Moscow's subway system is a model of speed, reliability, comfort, and affordability; it costs only 5 kopecks (8 cents Canada) to ride it in any direction and regardless of the number of transfers."

The First Subway Tunnel

When work began on the first Moscow subway tunnel in 1932, engineers and workers could hardly expect it would grow to be a gigantic and highly-automated operation in the course of less than 50 years. And neither could the 370,000 passengers who, three years later, rode the tiny 11.2 kilometre-long subway in Moscow's Sokolniki district on the first day of its operation, May 15, 1935.

Speed, Reliability, Comfort, and Cheapness, with 123 Subway Stops

Moscow's Metro system today, internationally acknowledged as the best in the world, is a model of speed, reliability, comfort and affordability; it costs only five kopecks (eight cents Canada) to ride it in any direction and regardless of the number of transfers. And many of the artistically decorated 123 subway stops along the 200 kilometre-long route resemble art galleries rather than a subway infested with graffiti in a metropolis of more than 8 million people.

Murals, Mosaics, and Semi-Precious Stones

Once below, commuters are often amazed to see they are surrounded by walls fitted with spectacular murals, frescoes, molded stucco, stained glass and other artistic features. Several stations are also decorated with statues and paintings by top Soviet artists, while many of the barrel-vaulted ceilings are adorned by Renaissance-style mosaics. These works of art are found among row upon row of multi-colored marble columns and pylons, hanging crystal chandeliers, and stairways and platforms made of granite, marble and semi-precious stones.

Expansion of the System

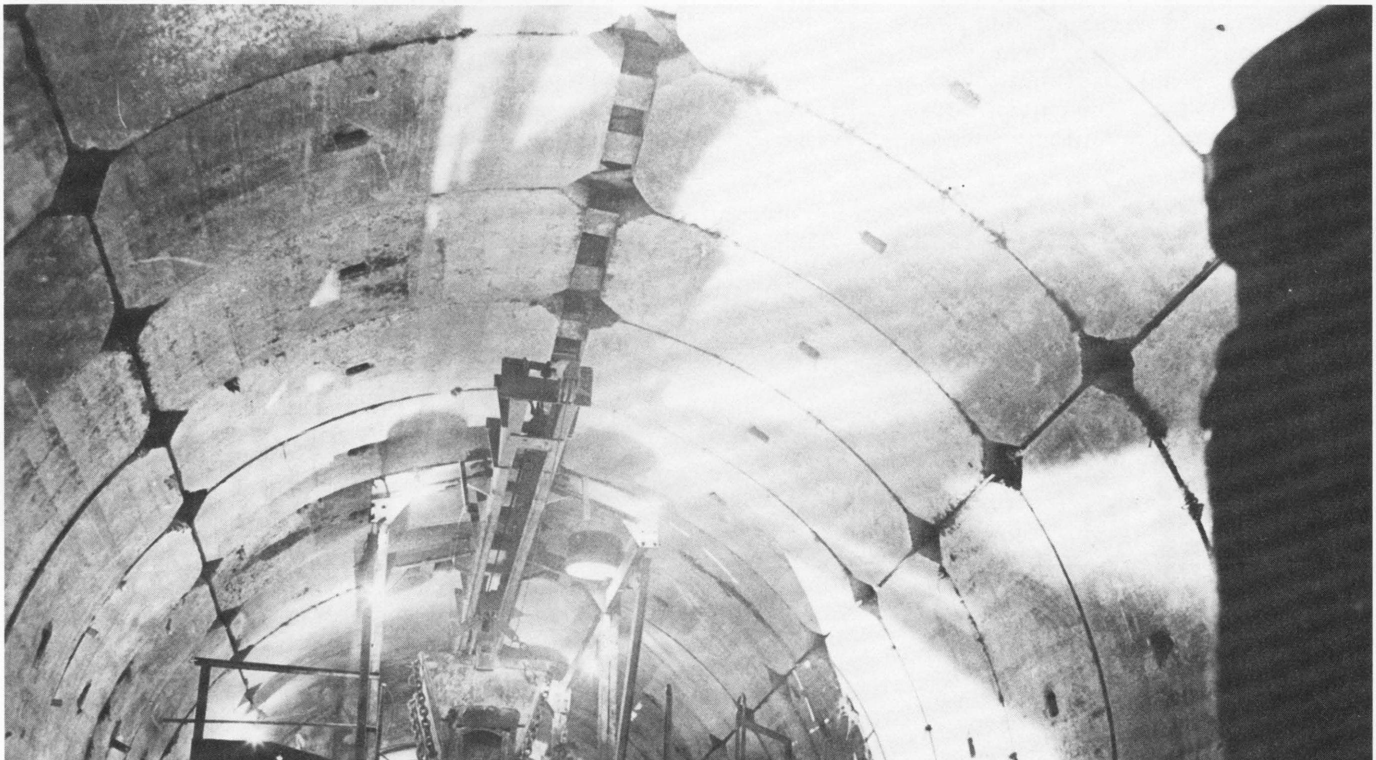
In the past three years, the Metro has expanded by an additional five kilometres along a new route, the Serpukhovsky line. It now links the capital's densely populated south-end with the city centre. By the end of the current decade, the Metro will expand by another 15 kilometres, thus bringing rapid transit to new areas now located beyond city limits. Plans are also under way to construct outer rings to facilitate transit among neighboring districts, thus eliminating the need for passengers going downtown in order to transfer back to a district neighboring their own. Besides, engineers are also considering the construction of rapid transit Metro lines that link Moscow's neighboring towns to the capital's centre.

Time Saving

Tunnel digging, building new rails and equipping Metros with trains and installations is a costly affair. Each new kilometre of Metro line costs nearly 15 million roubles ($24.7 million Canada). And with the current five kopek fare the Metro earns an average of only 17 million roubles a year ($28 million Canada). Yet it is a great money saver. On an average each underground trip in the Metro is nearly 18 minutes shorter than surface transport. In a year, the amount of hours saved totals more than 800 million, equalling nearly 500 million roubles ($825 million Canada). Consequently, Metro expenses are ultimately justified and paid for by the government.

Modernizing Transport Equipment

But planning and constructing new lines is only part of the problem. Modernizing existing lines and transport equipment are equally essential. The existing six-and-seven coach trains are currently being replaced by eight coach units to ensure higher transport capacity. Other modernization plans have also resulted in the gradual replacement of older rolling stock with new coaches made from lighter aluminum alloys equipped with internal noise-absorbing coatings. Mounted on pneumatically-sprung bogies, they provide smoother rides and highly dynamic characteristics in starting and braking. Trains made up of such coaches easily pick up speed to a maximum of 100 kilometres an hour.



Constructing metro tunnels and installing new rails, stations and equipment is very costly: each new kilometre-long line costs nearly 15 million roubles ($24.7 million Canadian). But moving fast saves Muscovites a total of 800 million hours a year, which translates into a saving of 500 million roubles.

Highly Automated Computerized Systems

The Moscow Metro now is a highly-automated precision system operating like a conveyor belt transferring people back and forth at a rate of nearly 8 million people a day.

A commuter stepping on a Metro escalator going at a speed of about one meter per second, may calculate the duration of the entire trip over any distance down to a minute. Recently, two computerized systems have been installed. They automatically regulate spacing and speed in such a way that trains follow each other in one minute intervals at an average speed of 45 kilometres an hour.

During the Metro's 19 and one-half hours of daily operation (6 a.m. to 1:30 a.m.), automatic devices ensure optimum speeds and frequency schedules to keep trains within safe limits of each other. As soon as trains come perilously close, the computerized systems immediately size up the situation -- without human intervention -- and slow them down.

Entering the Metro in Moscow's Pushkinskaya Station, commuters are transferred deep below along escalators on the fastest escalator system in the world.

Part of the Metro's more than 20,000 workforce are drivers. They merely stand by the 'wheel' checking and monitoring the controls, but ready to intervene in cases of emergency.

Eight More Soviet Cities with Subways

Soviet subways have begun in:

Leningrad

Kiev

Kharkov

Tashkent

Yerevan

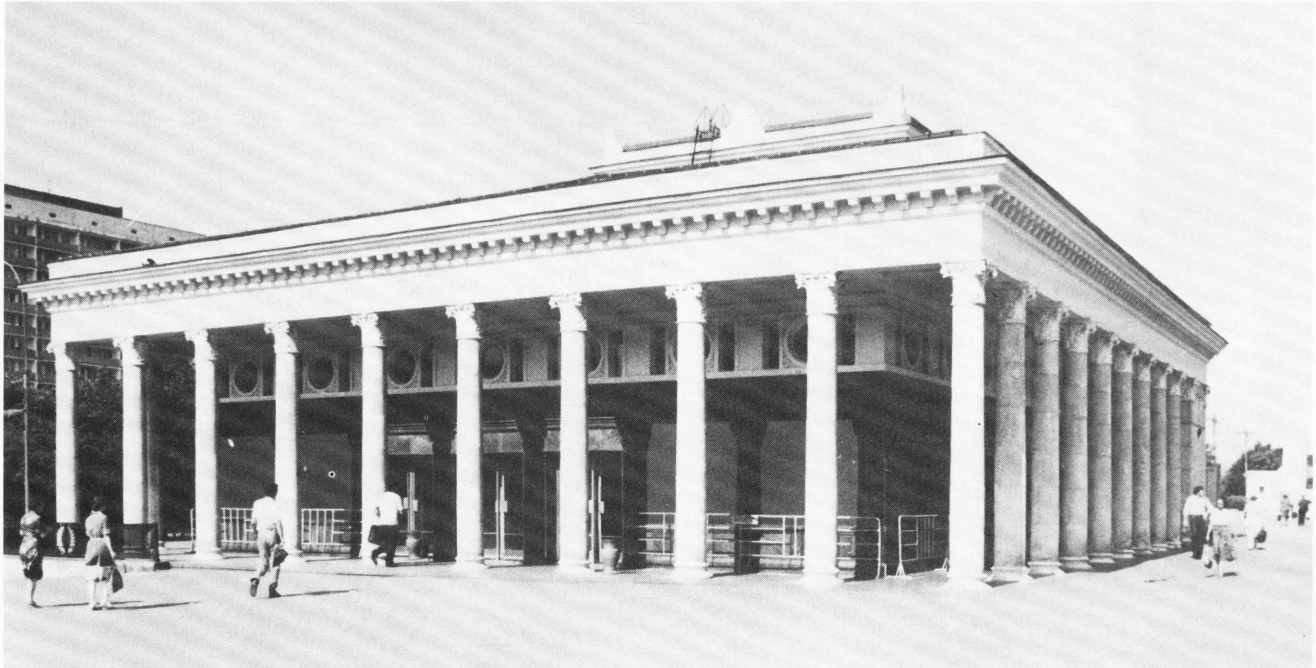
Tbilisi

Baku, and

Minsk

Computerized systems have been or are being installed in all these subways as well.

Passengers using Moscow's Metro system find reliable, fast, and comfortable service. Trains run in one minute intervals from each other at an average speed of 45 kilometres an hour, thanks to the recent installation of two computerized systems monitoring train spacing and speed.



The Dinamo Metro station entrance, designed by architect D. Chechulin, was built in 1938 during the fourth year of the Metro's operation.

Ω

Computers and the Japanese National Railway

L. Fletcher Prouty
Consultant
4201 Peachtree Place
Alexandria, VA 22304

"In this most modern railroad we see incorporated the concept that the railroad itself is a form of automated system that is enhanced by the broad application of computerized automation."

Planning Before World War II

As far back as the late nineteen-thirties the Japanese realized that their heaviest traveled routes were becoming saturated. A scheme was worked out to build a standard gauge rail line -- most Japanese trains run on narrow gauge -- to accommodate high-speed "Bullet" trains from Tokyo along the spine of Japan's main island, Honshu; and then by undersea tunnel to Hakata on the southernmost island, Kyushu. Planning was well underway and some construction had begun when further work had to be set aside during World War II.

Starting Again: The Tokaido Line

After the war, the Japanese National Railway (JNR) organization was re-established under Gen. Douglas MacArthur during the Occupation era. One of the first things done at that time was the electrification of the "Tokaido" line from Tokyo to Hakata. As this line became more efficient, it became saturated. It was carrying 130 trains per day, each way. By the mid-fifties the JNR had the good fortune to have been placed in the hands of the very able Mr. Sogo. He insisted on the construction of a separate line, with standard gauge, as had been planned before the war. He persuaded the government to go along with this plan and secured a loan from the World Bank. Five years after the start of construction the 320 mile segment between Tokyo and Osaka was opened in 1964, in time for the start of the 18th Olympic Games that were held in Tokyo. [Consider by comparison that the Washington, D.C. METRO system that dates back to 1954 has yet to complete its planned 100 miles.]

The Speed Decision: 130 Miles Per Hour

During these years of construction a segment of the new line was opened as a test site. The new all-electric trains were tested there and attained a speed of 178 mph. Encouraged by these achievements the JNR decided then on commercial operation at the top speed of 130 mph. They decided to build an all-new road-bed and track and at the same time to eliminate all crossings at grade throughout the entire high-speed system.

This speed decision was most important and had to come first. A modern, second generation railroad must be designed and constructed to meet a certain speed plan. It is possible to roll a steel wheel on a steel rail at 500 mph; but, does it make sense? Two solid wheels fixed on a solid steel axle would have to be manufactured so precisely to operate at 500 mph that it would not be worth the effort and cost. Of course the track would have to be a very precise bit of steel sculpture too.

Also, the traction power needed to overcome aerodynamic drag alone increases as the cube of the speed -- a very heavy penalty to pay for a very small saving in time. Furthermore, track construction and maintenance costs increase rapidly with speed, and what would such speed increases get you?

A train stops frequently to serve nearby markets. In fact, since so many communities were founded originally as "train stops," they have already been spaced that way. Therefore a 300 mph train would save very little time on a 30 mile segment over a 200 mph, or even 150 mph train.

The "High Speed Rail" Speed Zone

For these practical reasons it is now quite generally accepted that the "high-speed rail" speed zone is that between 115-175 mph. At least this is the regime that makes sense, and is financially viable. Actually the earlier "Bullet" trains were programmed to maintain an average speed of about 100 mph, including stops.

Thus the speed decision, or more properly the elapsed time decision, was fundamental. Once that had been addressed the engineers began to design the route. They kept it as straight and level as possible; but in the mountainous terrain of Japan that was not entirely possible. Therefore, in the 664 mile route between Tokyo and Hakata there are 217 miles (33%) in tunnels and 67 miles (10%) on bridges. This premier type of construction is costly, up front. In the long run it has saved money. Every single curve on the route was designed and constructed with a specific degree of superelevation (tilt) to permit the trains to maintain as uniform a rate of speed as possible. The best way to create a good average high speed is to eliminate slow speed sectors and bottlenecks. In this manner the road-bed and track systems were standardized and simplified for one type of operation.

Multiple Unit Motor System

At the same time other elements of the system were being developed. The "Bullet" trains have adopted an electric multiple-unit motor system that eliminates the locomotive with the advantages of reducing axle loads, of superior acceleration and deceleration and of reduced turnaround time. These sleek trains, standardized at 16 cars, in two car paired units, have relatively small electric motors installed on each axle and all are controlled by an on-board computer.

Computers to Eliminate Errors

As these were being developed, others were designing the Automatic Train Control (ATC) system to eliminate the possibility of the careless, human neglect of signals. The control over power, maintenance and passenger information, as well as information on the entire fleet of trains on the line is performed by a computerized Centralized Traffic Control (CTC).

All of these items are distinctive characteristics of the all-new, second generation, high-speed railroad system as demonstrated by the Japanese Shinkansen. In order to further modernize the system, computers have been even more widely introduced. As far back as 1977 the booking office could process requests for one million seats per day. The original traffic control system was upgraded to a COMTRAC (Computer-Aided Traffic Control) for control of the train fleet. Car inspection and repair are performed with the use of computers.

Automated Track Inspection

The inspection of tracks is performed by means of a train of high-speed track inspection cars that has proved most successful. The heart of the track inspection unit that runs on the same schedule as revenue trains is an inertial gyroscope linked to a computer. With this unit track tolerances are measured to millimeter units (no more than 7mm in any instance) and maintenance is ordered immediately in the event of any non-standard variation. The entire high-speed system is under the overall scrutiny of the Shinkansen Management Information System (SMIS).

The Reservation System

These are the general applications of the computer on this second generation high-speed rail system. In some ways they are over-shadowed by certain specific uses. For example: the first seat reservation system (MARS), was put in practical operation for JNR in 1960. The newest development of this massive system is MARS-301. This new system integrates the former MARS-105 comprehensive reservation system, the MARS-150 telephone reservation system and the MARS-202 group reservation system into one enormous system using one of the largest computers in Japan -- the HITAC M-280H. This new system not only unifies these three former systems but it makes maintenance easier and provides additional capacity for future expansion. The present MARS designation has been changed to mean Multi Access Reservation System.

This new system permits the terminal equipment to perform the following tasks:

(1) Expands the range of tickets being sold,

(2) Speeds up the ticket vending process,

(3) The terminal is more compact and improves operational functions,

(4) Provides information in characters along with a map diagram on each ticket issued,

(5) Provides magnetic encoding, automatic cancellation and automatic counting of each ticket,

(6) Contributes to overall labor savings throughout the railroad business, and

(7) Provides flexibility for expanding functions in the future.

The terminal station may contain up to six Operation Consoles, a Ticket printer and Journal printer. These are all located in the booking office and if the distance between them is short, less than 1.5Km, they are connected by an ultra-speed line (1Mbps).

Net Profit Per Year, More Than One Billion Dollars

In this most modern railroad we see incorporated the concept that the railroad itself is a form of automated system that is enhanced tremendously by the broad application of computerized automation.

The impact of such developments is enormous. The cost of labor on American railroads is between 50 and 60 percent of gross expenses, whereas on this high-speed Shinkansen line the labor cost is only 18 percent. Because of the liberal application of computer technology, productivity per employee is much higher.

By the end of the seventies this single high-speed rail system, in Japan, produced a verified net profit of more than one billion dollars per year that contributed more than 36% of the total cost of the operation of the entire JNR system. As reported by Mr. Tatsuya Ishihara, formerly Deputy General Manager of the Shinkansen Administration, "This financial success lies in the fact that it is an automated railway necessitating relatively few employees."

Safety Record: 2 Billion Passengers, Not One Casualty

The same can be said about its outstanding safety record of carrying about 2 billion passengers during a period of twenty years without a single casualty ... not one.

Extensions of Shinkansen

This initial route has been extended north to Niigata and northeasterly to Morioka. The route to Niigata is called the Joetsu and is 168 miles long. This route opened on November 15, 1982 and has averaged 43 trains per day. The route to Morioka is called the Tohoku and is 289 miles long. It has averaged 49 trains per day since its opening on June 23, 1982. Both new routes, through regions of lesser population density, have been successful far beyond original expectations.

Eventually, the network will be extended to the northern island, Hokkaido, via the longest undersea tunnel in the world; and to the central island, Shikoku, via a system with two of the longest suspension bridges on earth. Additional segments of the Shinkansen will link all major population centers of Japan.

The Shinkansen has now reached the status of one of the country's greatest assets and a National Treasure. It is one of the finest examples of man's creativity. There is no reason why such a system can not be duplicated anywhere on earth and why, with computers, that rail transport can not reach the peaks that it is capable of doing.

Japan has modern airways, super-highways and the traditional waterway links to all islands; yet it relies upon its excellent high-speed rail passenger system for the regular, safe and economical mass movement of millions of people. The Shinkansen is the nation builder.

Lesson for America

This is a lesson we need to learn in America. Our railroad network may have been built too early. It peaked and is now at a low ebb. Our highest authorities have warned of the coming transport crisis. Highways are congested and airways and airports are overcrowded yet the demand for more and better transport increases. There is no more space to meet the massive demands of highway development where it is needed the most, and land for new jetports just does not exist. For example, there is far more acreage in Kennedy International Airport in New York than all the acreage in the railway between New York City and Chicago. Something must be done to provide an alternative system to meet the growing demand. It will not be met by highways and airports. Transport at its best is a mix and the best mix in a modern first-tier country has good railroads at its core.

It is entirely possible to run freight trains on a high-speed rail system such as the Shinkansen. The only thing that must be remembered is that they must run at the same speed as the passenger trains, and they must not be too heavy. The technology does not mind what is on board the cars as long as they all operate the same way.

Testimony of Alan S. Boyd

Several years ago, Alan S. Boyd, then the president of the National Rail Passenger Corporation (AMTRAK) and formerly the Secretary of Transportation, Chairman of the Civil Aeronautics Board and one-time president of the Illinois Central Railroad Co. traveled to Tokyo at the invitation of Mr. Fumio Takagi, president of the Japanese National Railways. While there Mr. Boyd had the opportunity to ride the Shinkansen, to study its entire system and, in particular, to take a good look at its financial situation. He was pleased and properly impressed by all that he saw.

A few days after his return from Japan, he summed up his views of high-speed rail passenger services in a speech delivered before a railroad group in Chicago:

"I emphasize the Shinkansen 'system' because that is the most accurate way to describe it and to understand why it is different. The Shinkansen rolling stock is impressive, but there are very impressive trains in other countries. The Shinkansen right-of-way, roadbed, and track are impressive, but certain passenger rail sections in other countries may be as well designed. The Shinkansen central traffic control, computerization, and management are excellent, as they are in other countries. Its maintenance is extraordinarily efficient, and some other countries may match that. It is when you put all of this together into a fully integrated system that you discover, beyond all doubt, that the Shinkansen system has no equal. It is well designed, well organized, well supported, and well managed."

Future High-Speed Rail Systems

If high-speed rail systems are ever to be developed in the United States, two things must be made clear. First, new roadbed and tracks must be built. Passenger service and freight service must be handled with the same speed regime if they use the same track. Second, the construction of new track must be done as part of an integrally designed system -- i.e. track, vehicles, controls, stations, and operating practices with all their interfaces throughout the entire system. Then, the entire system and all of its sub-systems, must be computerized. None of these problems is insurmountable as other nations have demonstrated. In fact, other nations have shown that the future of high-speed rail technology is already waiting for us. We have to understand how to design, build and operate it.

It has been proved that high-speed rail systems can be profitable. If such systems are to be built in America, they are going to have to be owned and operated by a "for profit" private industry that knows what it is doing. No such system will ever be built successfully in the United States by a local, state or federal government or instrumentality thereof. This is a free enterprise profession, and it runs best that way.

Ω

U.S. Postal Service

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION

Required by 39 U.S.C. 3685)

1A. TITLE OF PUBLICATION: Computers and People

1B. PUBLICATION NO.: 0 3 6 1 1 4 4 2

2. DATE OF FILING: Oct. 1, 1984

3. FREQUENCY OF ISSUE: bi-monthly

3A. NO. OF ISSUES PUBLISHED ANNUALLY: 6

3B. ANNUAL SUBSCRIPTION PRICE: $18.50 domestic, $25.50 foreign

4. COMPLETE MAILING ADDRESS OF KNOWN OFFICE OF PUBLICATION *(Street, City, County, State and ZIP Code) (Not printers)*: 815 Washington St., Newtonville, Middlesex County, MA 02160

5. COMPLETE MAILING ADDRESS OF THE HEADQUARTERS OF GENERAL BUSINESS OFFICES OF THE PUBLISHER *(Not printer)*: same as item #4

6. FULL NAMES AND COMPLETE MAILING ADDRESS OF PUBLISHER, EDITOR, AND MANAGING EDITOR *(This item MUST NOT be blank)*

PUBLISHER *(Name and Complete Mailing Address)*: Berkeley Enterprises of New York, Inc. - same address as item #4

EDITOR *(Name and Complete Mailing Address)*: Edmund C. Berkeley, 815 Washington St., Newtonville, MA 02160

MANAGING EDITOR *(Name and Complete Mailing Address)*: same as above

7. OWNER *(If owned by a corporation, its name and address must be stated and also immediately thereunder the names and addresses of stockholders owning or holding 1 percent or more of total amount of stock. If not owned by a corporation, the names and addresses of the individual owners must be given. If owned by a partnership or other unincorporated firm, its name and address, as well as that of each individual must be given. If the publication is published by a nonprofit organization, its name and address must be stated.) (Item must be completed.)*

FULL NAME	COMPLETE MAILING ADDRESS
Berkeley Enterprises of New York, Inc.	815 Washington St., Newtonville, MA 02160
Edmund C. Berkeley	same as above
James J. Cryan	33 Sandra Lane, Wayne, NJ 07470
Ed Burnett	40 East 10 St., New York, NY 10003

8. KNOWN BONDHOLDERS, MORTGAGEES, AND OTHER SECURITY HOLDERS OWNING OR HOLDING 1 PERCENT OR MORE OF TOTAL AMOUNT OF BONDS, MORTGAGES OR OTHER SECURITIES *(If there are none, so state)*

FULL NAME	COMPLETE MAILING ADDRESS
none	

9. FOR COMPLETION BY NONPROFIT ORGANIZATIONS AUTHORIZED TO MAIL AT SPECIAL RATES *(Section 423.12 DMM only)* The purpose, function, and nonprofit status of this organization and the exempt status for Federal income tax purposes *(Check one)*

☐ (1) HAS NOT CHANGED DURING PRECEDING 12 MONTHS ☐ (2) HAS CHANGED DURING PRECEDING 12 MONTHS *(If changed, publisher must submit explanation of change with this statement.)*

10. EXTENT AND NATURE OF CIRCULATION	AVERAGE NO. COPIES EACH ISSUE DURING PRECEDING 12 MONTHS	ACTUAL NO. COPIES OF SINGLE ISSUE PUBLISHED NEAREST TO FILING DATE
A. TOTAL NO. COPIES *(Net Press Run)*	3500	3500
B. PAID CIRCULATION 1. Sales through dealers and carriers, street vendors and counter sales	0	0
2. Mail Subscription	2960	2863
C. TOTAL PAID CIRCULATION *(Sum of 10B1 and 10B2)*	2960	2863
D. FREE DISTRIBUTION BY MAIL, CARRIER OR OTHER MEANS SAMPLES, COMPLIMENTARY, AND OTHER FREE COPIES	54	46
E. TOTAL DISTRIBUTION *(Sum of C and D)*	3014	2909
F. COPIES NOT DISTRIBUTED 1. Office use, left over, unaccounted, spoiled after printing	486	591
2. Return from News Agents	0	0
G. TOTAL *(Sum of E, F1 and 2—should equal net press run shown in A)*	3500	3500

11. I certify that the statements made by me above are correct and complete — SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER: Edmund C. Berkeley, Editor

PS Form 3526 July 1982 *(See instruction on reverse)*

Computing and Data Processing Newsletter

COMPUTER-SIMULATED ROBOTS

*Based on a report in the "Financial Times",
Sept. 1984
10 Cannon St.
London EC4P 4BY, England*

Two investigators at Nottingham Univ., England, believe that they have produced a system which could reduce the hazards of installing robot systems in industry.

They have developed a computer program called GRASP -- Graphical Robot Applications Simulation Package -- which can simulate robots at work in a factory, the idea being to cut the time to design flexible manufacturing systems and reduce mistakes.

GRASP cost about £250,000 to develop, of which £150,000 came as a grant from the Science and Engineering Research Council.

Dr. Yoon Yong and John Green of Nottingham University are two of the founders of a new company, BYG Systems, created to market the system. Dr. Yong and his fellow workers moved into factory space on Nottingham University's new science park in October.

Dr. Yong believes that there will be two major applications for GRASP, the design and layout of new factory automation systems and as a programming aid for robots in existing systems.

The Nottingham University researchers have worked closely with industry during the program's four year development. Computer instructions to the designer are given in clear English to avoid the need for computer jargon. GRASP has simulated a group of four robots welding a truck for Ford, for example. If a designer tries to make a robot perform a move of which it is not capable, GRASP will show this on a computer screen. It can also indicate if a robot is likely to hit an obstacle as it moves.

With GRASP, designers have a pre-defined set of basic shapes from which to build up robot models on the computer screen. They can simulate all the leading makes from Asea, Unimation, Cinncinnati Milacron or IBM, for example. Workpieces, benches and conveyors can be added to build up a picture of the proposed factory environment.

These are three dimensional models which can be moved around the screen at will, seen from any angle, and the effect of moving one component can be assessed. Animated sequences are possible where the simulated robot is put through its paces as if it were a properly programmed machine on the factory floor.

Examples could include a robot picking items off a conveyor or presenting an engineering part for drilling or polishing. GRASP can even simulate the time taken for a real robot to do the job thus highlighting any synchronization problems when linking machines on the production line.

Once a work sequence is programmed, a robot of one make can be substituted for another. Technical specifications such as degree of rotation, length of a robot's arm, speed of movement are all entered into the computer. Dr. Yong said that robots could thus be compared for their suitability for a particular job. "This can stop you buying an overly-sophisticated robot when a simple one will do," he said.

GRASP will cost from £45,000 at which price it is the cheapest robot simulation system on the market capable of animation and identifying problems. Dr. Yong says that it brings the system within the range of small and even medium-sized companies.

NETWORK OF 400 PERSONAL COMPUTERS WITH LISP LANGUAGE

*Robert M. Byers
News Office
Mass. Inst. of Technology
Cambridge, MA 02139*

Massachusetts Institute of Technology (MIT) and Texas Instruments, Inc., (TI) have entered into an agreement under which the MIT Laboratory for Computer Science (LCS) plans to acquire up to 400 of TI's newly-developed Explorer personal computers. Over the next two years, these machines will be used to establish the world's largest network of Lisp machines, significantly changing the computational environment of the Laboratory.

The Explorer is an advanced personal computer which uses the LISP language and incorporates several major developments originated at M.I.T. The 400 machines are expected to have a value of some $20 million. "In exchange for initially developing the Explorer basic technology, M.I.T. expects to acquire the full system at a favorable price," Dr. Michael L. Dertouzos, LCS director, said. Funding to acquire the Explorer system will be sought from private sources and from the Advanced Research Projects Agency of the Department of Defense which funds somewhat more than half of the LCS's present research program.

"By 1986, we hope to have the entire laboratory -- faculty, students, staff and support staff -- using Explorers and linked together in a large interactive network," Dr. Dertouzos said.

For Dr. Dertouzos and his colleagues at LCS, the arrival of Explorers at M.I.T. will bring to the campus -- in one commercial product -- a whole series of computer science advances that began at M.I.T. These include: the LISP language developed in 1959 by Professor John McCarthy, now at Stanford University; the Nu Bus and Nu Machine technology developed at LCS in the late 1970s under the direction of Professor Stephen A. Ward; the Lisp machine and the notion of a total Lisp environment that started in M.I.T.'s Artificial Intelligence Laboratory in the late 1970s; and, finally, concepts and techniques for interconnecting hundreds or thousands of machines into a distributed system, a key concept which LCS has been pursuing since 1976.

"All our ideas are coming back to us in one well-engineered commercial system," Dr. Dertouzos said. "Putting all these developments together and adding their own considerable contributions was a challenge for TI and we think the company should be congratulated. It is something we have been hoping for for eight years." The Explorers will form LCS' central computational resource -- "the dominant keyboards and screens in front of our students, faculty, staff and support staff." "Explorers will produce documents, develop programs, handle mail and messages and give everyone access to external networks and to our own unique facilities -- our multiprocessor emulation facility, our very large scale integrated circuit design system, a number of laser printers and our data archives," he said.

"We will, of course, continue research on a variety of hardware and software systems outside of the Explorers -- on machines of other manufacturers and machines of our own design. But we will rely on the Explorers to access these machines and to carry out tasks that we presently accomplish through central time-shared and dedicated personal machines."

Dr. Dertouzos said the Explorers, when installed and linked together, will solve what he has called the "Tower of Babel" problem within LCS. Using a large number of different computers, LCS workers have found sharing research and interacting with each other difficult, he said.

LCS earlier received 30 prototype Nu Bus personal computers from TI after TI received the Nu Bus license from M.I.T. The aim was to begin weaning LCS from dependence on time-sharing, a concept LCS itself pioneered 21 years ago. The Nu machines sidestep obsolescence, thanks to the special communication standard called Nu Bus which makes possible the use of different processors as they evolve.

"The new Explorers are precisely in that mold," Professor Dertouzos said. "They are the Nu machines to which Lisp processors developed by TI have been added." "Unlike a time-shared system that can have at most 50 or 100 users, a network of Explorers will be able to scale up to an arbitrarily large number of users. Nevertheless, users will see the same services -- with the same or better performance -- as they now see on a single central machine.

"The revolutionary aspect of distributed systems is the transition from several centralized single-computer environments to a large computer 'tribe' of decentralized -- but equally powerful -- intercommunicating machines." Nearly half of LCS's 320 researchers already are doing research on distributed systems, Professor Dertouzos said. "Our approach to distributed systems," he said, "has been a search for equilibrium between the opposing forces of cohesiveness and local autonomy -- cohesiveness because applications such as sharing common data bases require a single coordinated approach, and local autonomy because of people's inherent need to control and use their own resources for their own purposes. Decentralization will lead to acceptable operation of an aggregate system in spite of local failures."

LCS was started in 1963 as Project MAC (Multiple Access Computer and Machine Aided Cognition) with ARPA support. Compatible Time Sharing System (CTSS), one of the first time-shared systems in the world, was an early laboratory success. In the 1960s, the laboratory developed Multics, an improved time-shared system introducing virtual memory, operating system security and the writing of operating systems in a high-level language -- developments that spurred the application of on-line computing to engineering, architecture, mathematics, biology, medicine, library science and management.

COMPUTER-CONTROLLED ROBOT TO SENSE THE ENVIRONMENT BEING DESIGNED AT UNIV. OF PENNSYLVANIA

Virgil Renzulli
Univ. of Pennsylvania News Bureau
410 Logan Hall
Philadelphia, PA 19104

A machine in a University of Pennsylvania research lab can use sight and touch to identify objects such as cups and saucers and can pick up eggs and lightbulbs firmly enough not to drop them but gently enough not to crush them.

The machine is part of a research project to develop computer-controlled devices that can sense, manipulate and understand their environment, devices that will become the household and industrial robots of the future. The work is being done at the General Robotics and Active Sensory Perception Lab, directed by Dr. Ruzena Bajcsy and Dr. Samuel Goldwasser of Penn's School of Engineering and Applied Science.

Bajcsy and Goldwasser and their colleagues are working on both the hardware and software to build intelligent robots that will be able to sense and adjust to their environment. In the future such robots would be able to do delicate jobs both in the workplace and in the home, such as sorting and testing parts on an assembly line, remotely piloting a vehicle or setting the dinner table.

In addition to its use in robotics, their software developments have led to new types of 3-dimensional computer imaging. For example, a two-dimensional scan via computer-assisted tomography of a living patient's skull can be processed to create a three-dimensional model that can be cut open to reveal the skull's internal structure. Although still in the prototype stage, the imaging system is expected to have important applications in clinical diagnosis and surgical planning.

The robots commonly used in industry perform very specific and limited tasks under programming that is almost totally fixed. Such a robot would flunk a simple household task that required flexibility, such as setting a dinner table. However, the researchers at Penn have developed a robotics system that will enable a robot to distinguish, for example, a coffee cup from a wine glass or a soup bowl and then pick up each with the appropriate grasp. Since there are a variety of grasps, the particular one used by the machine can indicate whether the object was properly identified.

Bajcsy explained that developing a computer vision system required a thorough understanding of how humans see and that it was necessary to study what researchers in psychology, physiology and other fields have learned about human vision. The robotics system she and her colleagues have developed has a computer data base containing images of real world objects. Each object, such as a cup or a bowl, was reduced to its most fundamental and unchangeable characteristics; the difficult part was eliminating the nonessential characteristics while keeping everything that is vital to the object's identification. Rather than have the computer compare a perceived object to every image it has stored, they designed software that enables the computer to compare the perceived object to a small field of representations in its data bank.

In addition to the software developments that have provided the brains for the robotics system, the researchers have also come up with innovative hardware designs for vision and tactile sensing. The robot vision system consists of two cameras with motorized zoom, focus and iris controls, mounted on a motorized platform and combined with a lighting system. This allows the computer the same flexibility that a person has to view an object from several angles, move closer to it or change the lighting on it.

"The camera will move to a particular point and take a picture," says Goldwasser. "The program will analyze it and determine whether or not the picture contains the object the computer is interested in. If not,

(please turn to page 27)

Borning – *Continued from page 11*

The Limits of Feasible Testing

To be at all confident of the reliability of complex systems, there must be a period of testing under conditions of actual use. Simulations, analyses of possible modes of failure, and the like can all expose some problems, but all such tests are limited by the fact that the designers test for exactly those circumstances which they anticipate may occur. It is the unexpected circumstances and interactions -- an unexpected succession of lightning strikes, mechanical failure followed by operator error, a fire that destroys several cables simultaneously -- that cause the most severe problems. Short of having a nuclear war, the command and control systems for nuclear forces cannot be tested completely.

A final issue is that systems in flux are more prone to problems than those which have remained stable for some time. If the arms race continues unabated, the command and control systems for nuclear forces will necessarily be changing as well.

Technological Solutions Are Not a Viable Answer

To summarize, at present a nuclear war caused by an isolated computer or operator error is probably not a significant risk, at least in comparison with other dangers. The most significant risk of nuclear war at present seems to come from the possibility of a combination of an international crisis, mutually-reinforcing alerts, and possible computer failure and human error.

A continuing trend in the arms race has been increasingly accurate missiles. The development of such missiles, and particularly their deployment in locations close to enemy territory, leave less and less time to react to and evaluate an alert (and determine whether it is real or due to a computer or human error). People simply cannot make decisions of such scope and consequence in a matter of minutes. Thus we see increasing pressure to consider a launch-on-warning strategy where we turn the decisions over to computers. But I have argued here that such a strategy seems foolhardy -- that computers are no better at such decisions than the programmers who write their software.

Where then does that leave us? The conclusion is clear: technological solutions to the problems of the arms race, such as more accurate weapons or more sophisticated computer hardware and programs, are not a viable answer. They may give the illusion of providing a solution, but the illusion is extremely dangerous. The problem is, in the end, with the policies and relations between governments that have created the threat that is now pushing us to make such unreasonable demands on technology. It is these policies and attitudes that are the root of the problem, and it is in this human and political realm that solutions must be sought.

Ω

Newsletter – *Continued from page 26*

the camera will move to another place and look for it. If it is the object the computer is interested in, the computer might zoom in closer or adjust the angle of view slightly. Most of those adjustments are already under computer control; the camera's zoom, focus and iris controls will be added shortly."

The robotic vision is combined with tactile devices -- a mechanical hand and a sensor that is shaped like a rigid finger. The finger, which is mounted on a robot arm, contains 133 pressure-sensitive sites covered by a conductive foam. It can be used to trace the shape of an object, determine its surface texture, and find cavities in it, such as the hole and handle of a cup. If the computer were searching for a cup and found a cylindrical object that appeared to have a handle on its side and a hole in its top, it could use the finger to determine whether the apparent handle and hole were real or just shadows. (It could also change the lighting or angle of view to accomplish the same task.) Once an object has been identified, the computer can use a specially designed mechanical hand to grasp the object properly -- an easy task for a person but a very difficult one for a machine -- and then to manipulate it.

The robot hand has more dexterity than those commonly used in industry. It has three fingers, each with two joints and tactile sensors, and the third finger can move about the base to oppose the other two in a [illegible]nctioning of the human thumb. Because [illegible] hand is a more compl[illegible]e finger, it requires mo[illegible]ramming, but eventually it [illegible] finger on the robot arm an[illegible]uch sensing.

Ω

Games and Puzzles for Nimble Minds — and Computers

Neil Macdonald
Assistant Editor

It is fun to use one's mind, and it is fun to use the artificial mind of a computer. We publish here a variety of puzzles and problems, related in one way or another to computer game playing and computer puzzle solving, or to programming a computer to understand and use free and unconstrained natural language.

We hope these puzzles will entertain and challenge the readers of *Computers and People.*

NAYMANDIDGE

In this kind of puzzle an array of random or pseudo-random digits ("produced by Nature") has been subjected to a "definite systematic operation" ("chosen by Nature"). The problem ("which Man is faced with") is to figure out what was Nature's operation.

A "definite systematic operation" meets the following requirements: the operation must be performed on all the digits of a definite class which can be designated; the result must display some kind of evident, systematic, rational order and completely remove some kind of randomness; the operation must be expressible in not more than four English words. (But Man can use more words to express the solution and still win.)

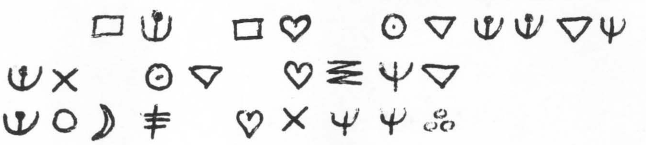
NAYMANDIDGE 8501

9	6	8	0	0	1	2	4	6	7	6	3	6	5	0	7	2	6	4	3
8	3	8	4	1	1	2	5	5	2	2	2	2	3	8	4	4	9	4	7
0	3	7	9	8	2	7	5	2	8	5	6	3	4	4	3	6	4	9	3
6	0	5	5	0	8	5	9	4	4	2	3	4	0	2	4	1	3	9	5
5	0	3	6	8	2	6	9	3	7	4	6	5	4	0	5	9	1	2	8
2	7	1	1	1	8	1	4	4	5	5	2	0	4	0	6	0	7	2	5
8	4	8	4	5	6	4	0	3	3	3	6	3	8	3	5	6	9	4	2
2	0	5	8	4	5	6	8	5	3	4	4	8	5	9	6	4	2	2	7
2	7	4	6	7	3	5	9	0	1	9	9	8	1	6	5	8	1	7	8
4	2	0	7	2	8	8	6	4	8	3	0	2	2	1	6	1	2	7	2

MAXIMDIDGE

In this kind of puzzle, a maxim (common saying, proverb, some good advice, etc.) using 14 or fewer different letters is enciphered (using a simple substitution cipher) into the 10 decimal digits or equivalent signs, plus a few more signs. To compress any extra letters into the set of signs, the encipherer may use puns, minor misspellings, equivalents (like CS or KS for X), etc. But the spaces between words are kept.

MAXIMDIDGE 8501



NUMBLES

A "numble" is an arithmetical problem in which: digits have been replaced by capital letters; and there are two messages, one which can be read right away, and a second one in the digit cipher. The problem is to solve for the digits. Each capital letter in the arithmetical problem stands for just one digit 0 to 9. A digit may be represented by more than one letter. The second message, expressed in numerical digits, is to be translated (using the same key) into letters so that it may be read; but the spelling may use puns or deliberate (but evident) misspellings, or may be otherwise irregular, to discourage cryptanalytic methods of deciphering.

NUMBLE 8501

```
          Y O U R
*           S O N
        ---------
        A O J I Y
      N O I A S
    A E S Y O
  ---------------
= A I S R E U Y
```

55323 67882 90

Our thanks to the following person for sending us solutions: T.P. Finn, Indianapolis, IN – Maximdidge 8407, 8409; Numble 8407.

SOLUTIONS

MAXIMDIDGE 8409: Life is a light in the wind.
NUMBLE 8409: First lay the egg, then cackle.
NAYMANDIDGE 8409: Column 5: 2, 3.

NAYMANDIDGE 8409

7	6	1	1	3	5	0	8	9	1	8	2	1	7	3	8	5	7	7	2
5	0	4	5	3	5	2	4	8	2	5	0	4	0	5	3	4	3	4	3
5	3	0	2	2	6	6	0	1	6	1	8	1	5	9	5	0	3	4	4
3	6	8	9	3	0	2	8	6	3	0	2	4	5	5	2	9	5	1	3
0	1	4	8	2	4	4	9	5	4	9	1	2	6	0	7	5	4	2	5
2	8	8	1	2	8	3	5	1	1	8	8	1	7	0	5	2	2	5	4
7	9	2	4	3	5	0	5	0	4	3	2	3	1	0	1	0	9	9	5
1	9	8	0	3	4	2	0	7	2	2	5	8	2	9	2	2	5	5	7
0	9	8	7	2	4	6	7	1	8	2	0	3	6	9	4	7	6	5	0
7	5	5	8	3	5	6	3	2	4	3	1	0	1	4	4	0	0	9	9