

越中のすすめ

Beyond Turing

—— チューリング計算限界を越えよう！
計算とコンピュータの黎明期へ戻って ——

ご注意；著作権に関係するスライドや部分は、削除されています。

星野 力

筑波大学名誉教授

計算科学研究センター・フェロー

チューリングという名前を知っていましたか？

まったく初耳

計算理論、チューリング・マシン

エニグマ暗号、チューリング・ボンベ

ACE コンピュータ

人工知能、チューリング・テスト

生物の形態形成、チューリング・パターン

ホモセクシャル

ACMチューリング賞

S F (クリプトノミコン)、映画 (U-571)、演
劇 (掟；劇団四季)

マンガ (ブレインズ コンピュータに賭けた男)

星野の著作 (甦るチューリング)、講義など

チューリング・イヤー (2012年、生誕100周年)

ALAN TURING YEAR



2012

なぜ、いまチューリング を回顧するのか？

- スーパーコンピュータはエクサの壁を超えることができるか？
- ハード的手段（プロセッサ単体の速度と並列台数）は限界にきている。
百万プロセッサなら、原発1基の電力が必要。
- 逐次処理の範囲内では限界を越えられない？
- チューリング計算限界を越えるには、大きなパラダイム変化が必至。
- 将来のコンピュータを創造する若い人たちは、
ぜひ

歴史感覚と根本に立ち返る問題意識をもって

チューリングって誰 (年表1)

- Alan M. Turing 1912年6月23日ロンドン生まれ。
- 大学院生 (1936年、24歳) のとき、コンピュータ (計算手: computer) の基本モデル「チューリング・マシン」を提唱。計算理論を創始。コンピュータ科学のパイオニア。
- 2次大戦中(1939~1945年)、ドイツ軍のエニグマ暗号をチューリング・ボンベによって解読し、多くの人命が大西洋の底へ消えるのを救った。イギリスの国家英雄

チューリング (年表2)

- 2次大戦後、汎用チューリング・マシンの実現を目指して、コンピュータACEの開発を主導した(1945~1947年)。
- マンチェスター大学(1948~1954年)の世界最初のコンピュータ (Baby & Manchester Mark-I) を駆使して、生物の形態形成 (チューリング・パターン) の計算をした (1952年発表)。
- 人工知能を深く考察し、チューリング・テストを提唱した (1950年)。

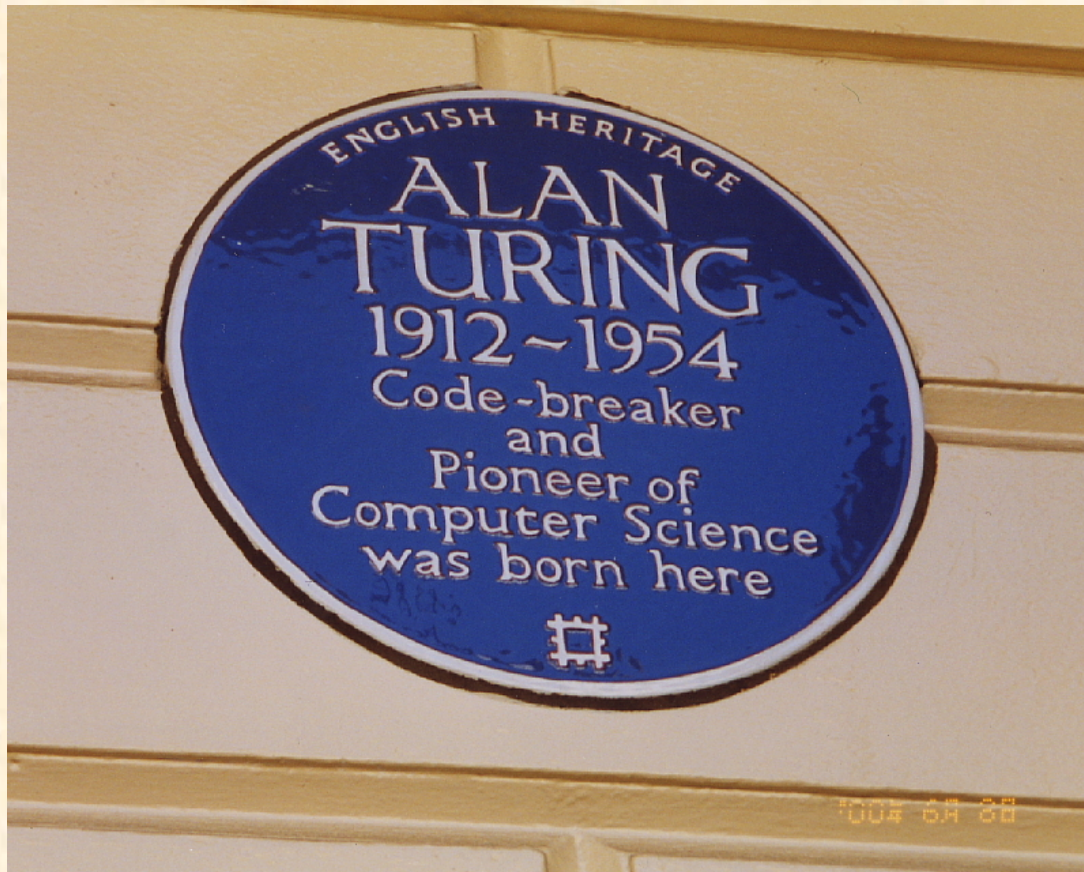
チューリング (年表3)

- マラソン・ランナー (1948年ロンドン・オリンピックには怪我で出場しなかった)。
- ホモセクシャルとして逮捕され有罪判決を受けた(1952年)。しばらくして謎の自殺を遂げた (1954年6月7日)。
- 計算理論はコンピュータ教育の基礎となり、ACMはチューリング賞を設けた (1966年)。
- 2009年英国首相は有罪判決を謝罪し、

マラソン競技会にて

生誕の地

(現在Colonnade Hotel)



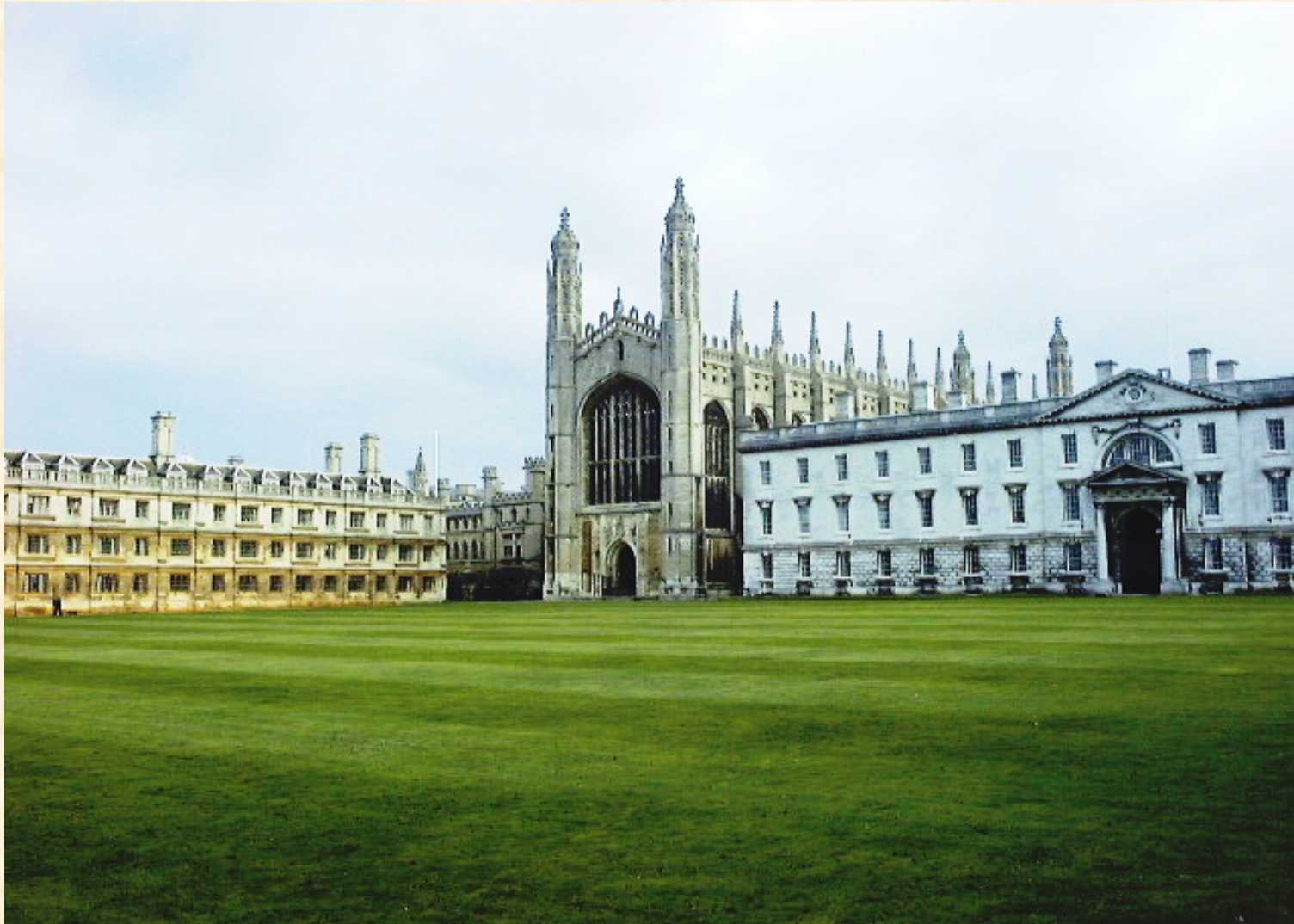
英国遺産の青プレート

パブリック・スクールの 不適應児

- 両親はインドへ赴任。退役軍人家庭へ預けられる。
- 1926年（14歳）シャーボーン・パブリックスクールに入学。成績低く不適應生徒。
- 「服装がだらしない」「もし科学の専門家になるつもりなら、ここで時間を無駄にしている」「どの学校でも社会でも問題児とみなされる類の生徒」。しかし、理系には強かった。
- 1931年（19歳）ケンブリッジ大学キングスカレッジに入学。

ケンブリッジ大学キングスカレッジ

この美しい芝生にはフェローしか立ち入れない



Tea Garden "Orchard"

世界一有名な(人が来た)喫茶店



**Bertrand Russell Maynard Keynes
Crick and Watson Wittgenstein
Stephen Hawking Sir J. Cockcroft
Whitehead Alan Turing Rutherford
Virginia Woolf A A Milne
King George VI Prince Charles
Pandit Nehru 他**

数学はいま危機に瀕している！

- ヒルベルトの檄(1900年)：数学を再構築せよ。公理主義・形式主義を提唱。
- ヒルベルトの23の未解決の問題の提唱。第10問題：ディオファントス方程式（例： $6X^3YZ^2 + 3XY^2 - X^3 - 10 = 0$ ）が整数解を持つか否かを判定する有限的手順をみつけよ。
- 1920年代になって、決定問題：「どの命題に対しても、それが正しいかどうかを決定する一般的な手順が存在するか」、具体的には「述語論理において論理式が充足可能かどうかを決定する有限の手順はあるのか」という問題。
- 1930年、クルト・ゲーデルによる否定的解答（不完全性定理）。

計算可能な数 について (1936年)

- 後にチューリング・マシンと呼ばれた
- 計算している人間 (computer)
- の心理と操作を模擬する機械的モデルを考えた。

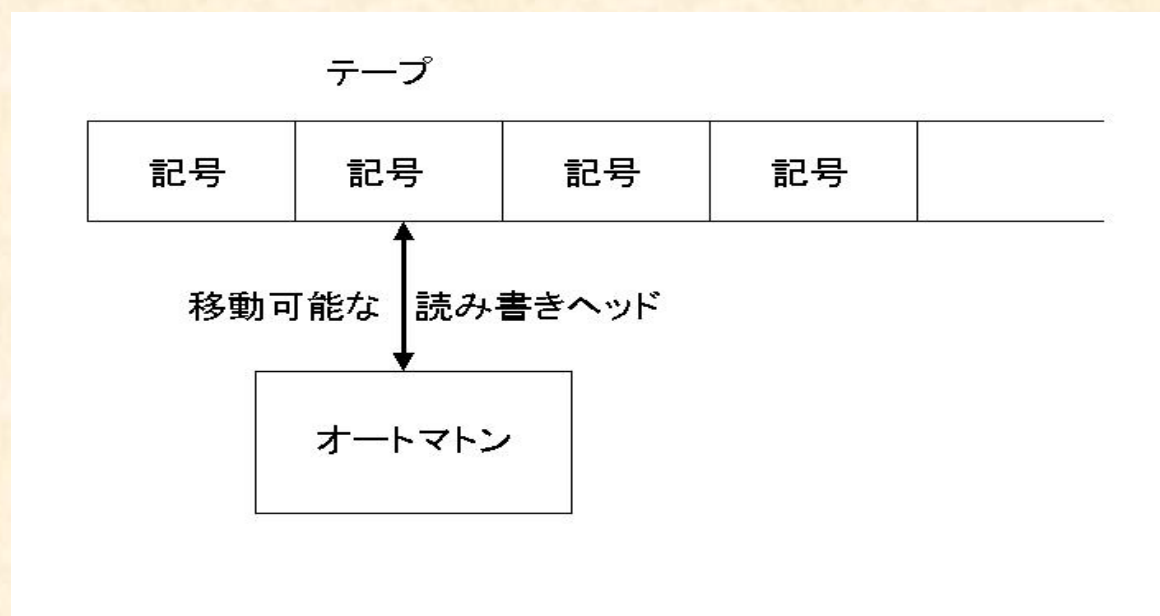
We may compare **a man in the process of computing** a real number to a machine which is only capable of a finite number of conditions The machine is supplied with a “tape”, (the analogue of paper) running through it, and divided into sections (called “squares”) each capable of bearing a “symbol”. ... The behaviour of **the computer** at any moment is determined by the symbols which **he** is observing. and **his** “state of mind” at that moment.

It is my contention that these operations include all those which are used in **the computation of a number**.

数 = 命題 = 形式的な操作 だから **the computation of a number = 数学**

チューリング・マシン

計算している人(計算手)の心理モデル
記憶=テープの記号



チューリング・マシンは専用ハードウェアで、問題毎に作らないといけない

チューリングマシンで
どう計算するか？

COS を SINに書き換える

7 に 1を加える

チューリング計算可能とは

計算とは

対象のシミュレーションによって、なんらかの数的な答をえること (by星野)。手段はデジタルとは限らない。

チューリング・マシン（数学者の思考をモデル化したもの）で計算できることが、計算。その条件は、

- まず問題が記号化され形式化されていること。
（世の中に、記号・形式化されないことは無数にある）。
- チューリング・マシン上で、有限の実行ステップで終わる計算手順（アルゴリズム）が存在する。

計算不能問題

- しかし、アルゴリズム（プログラム）が存在しない問題がある。
- たとえば、停止問題（と停止問題に帰着できる問題）
- すべてのプログラム M と入力 X を相手にして、停止するかどうかを判定するプログラム P （ M 、 X ）は存在しない。
- 存在すると仮定すると矛盾が起こる（対角線論法）。証明は省略。

停止問題に帰着できる例

プログラムが仕様を満たしているか？

ウイルスに感染しているかどうか？

学生Bが提出したプログラムは

学生Aのプログラムの盗作か？

- こういうことを計算前に判定する万能プログラムは原理的にできない。
- 特定のプログラム（たとえば流行中のウイルス）だけを相手にする判定プログラムは可能で、実際商売になっている。

IT社会はチューリングマシンという 小さな土台の上に載っている

チューリング
計算不能

Simulation &
Turing test

脳

チューリング計算可
能

Applications

Script

Compiler language

Utility

Software

Assembly code

Function

Machine code

Firmware

Hardware

生命
自己参照・増殖・進化

チューリングマシン →



アナログ計算機

チューリングの死後、計算可能限界の内外は

- 1960年以後、階層化という手段によって、計算可能な世界は、上（人間に近い方）へ向かって多様で巨大な発展を遂げた。
- 計算不能でも、可能領域へ数学的シミュレーションによって射影（モデル化）し、アルゴリズムの探求によってある程度は計算できる。
- 例：人工脳（人工知能）。その成否はチューリングテストにより判断されている。
- 計算可能な範囲内でも、計算困難な（計算の複雑さが大きいため膨大な時間を要する）問題も多い。

計算不能世界

- 計算不能な領域の研究は1960年頃から数理論理の研究と、ロマンティックなイマジネーション業界で扱われてきた。
- 脳と知能に関する哲学的考察も。脳（人工知能）はコンピュータでは計算できない（ゲーデル）。チューリング・マシンには計算不能な問題がある（数学の不完全性）が、脳はそれを考えることができるから。
- 脳の低次なレベルにおいて、量子力学的なメカニズムに知能の可能性を説く人もいる（ペンローズ）。

話を計算可能世界へ戻すと.....

万能チューリングマシン (Universal Turing Machine)

UTM=任意のチューリングマシン (TM) をプログラムで表現し、テープにおく。

UTMのオートマトンはインタープリターつきコンピュータ。

すべてのTMをこのUTM上でシミュレート (計算) できる=万能性。

UTMとは、~~計算可能な範囲~~ ^{計算可能な範囲} 一切に他ならない。その本質は。

任意のTMのテープ



インタープリター
つき
コンピュータ

汎用性と逐次性

汎用性(万能性)について

- コンピュータ（チューリング・マシン）の汎用性とは、計算可能な問題であればすべて計算できること。この「計算」とは、数学的シミュレーション。
- シミュレーションは現実（物理現象、etc）とは違う。
- 初期の開発者・理論的リーダーは、数学者（チューリング、フォン・ノイマン）だった。コンピュータは「数学者の作品」である。
- ハードウェアが高価な時代では、電気技術者はできるだけ少ない電子回路（と予算）で、できるだけ広範囲な問題を解きたかった。
- 汎用性は商業的に有利。しかし、至上命令ではない。

逐次性について

- チューリング・マシンは、数学者の思考のモデル。
- 数学の思考は（ヒルベルトの形式主義では）明示的・逐次的であって、直感的な飛躍的思考は排除されている。
- 逐次的思考は、入力・出力をもつ能動的論理素子を結合した、状態遷移する順序論理回路の上で実現されている。
- 逐次性は、電子回路から、論理回路、アーキテクチャ、プログラミング、アプリケーション、そして**人間の考え方**まで一貫して支配している。
- **逐次性はチューリング計算パラダイムの本所**

では、どうするか？

- 逐次性に起因する低速性は、チューリング・マシンの宿命。
- 逐次性を否定すると、チューリング・マシンではなくなり、汎用性がなくなる。
- 速度限界を超えるには、チューリング計算限界を越え、逐次的でない、あたらしい計算原理を見つけないといけない。
- その原理は凡人にはわからないが、**計算不能世界からのひとつのヒントはある？**

話を1937年に戻すと

留学生のチューリングは.....

- 計算不能な数の階層世界を論じる博士論文を指導教授のアロンゾ・チャーチ（プリンストン大学）に提出したが、
- 近くにいたフォン・ノイマンは、チューリングに助手になってアメリカに残るように勧めた(1937年頃)。
- しかし、彼はケンブリッジ大学キングスカレッジへ戻った(1938年)。
- 1939年9月、ヒトラーはポーランドへ侵攻し、イギリスはドイツに宣戦布告した。

暗号破りのプロフェッサ

- 1939年（27歳） 第2次世界大戦が始まった。
- ただちにチューリングは志願して、暗号解読に従事。ブレッチレイ・パークにおいて、エニグマ暗号解読装置ボンベを開発する。
- 「ウルトラ」と呼ばれた暗号解読プロジェクトは、戦後のヨーロッパの世界を決めた、イギリスのマンハッタン計画。

エニグマ暗号の詳細な
解読方法ではなく.....

チューリング・ボンベ（エニグマ暗号解読機）
で用いられた**アナログ的双方向結線（
恒等回路）**

による「計算」

にかすかな可能性がある??

- まず、エニグマ暗号と通信方法は？
- チューリング・ボンベによる解読方法は？

エニグマ暗号の特徴と欠点

- 反転性のある換字暗号

例 $a \longleftrightarrow k$ $e \longleftrightarrow w$

対合 (involution) という;

$$\text{inv } f(x) \equiv f(x)$$

平文	a	e	...	k
暗号文	k	w	...	a

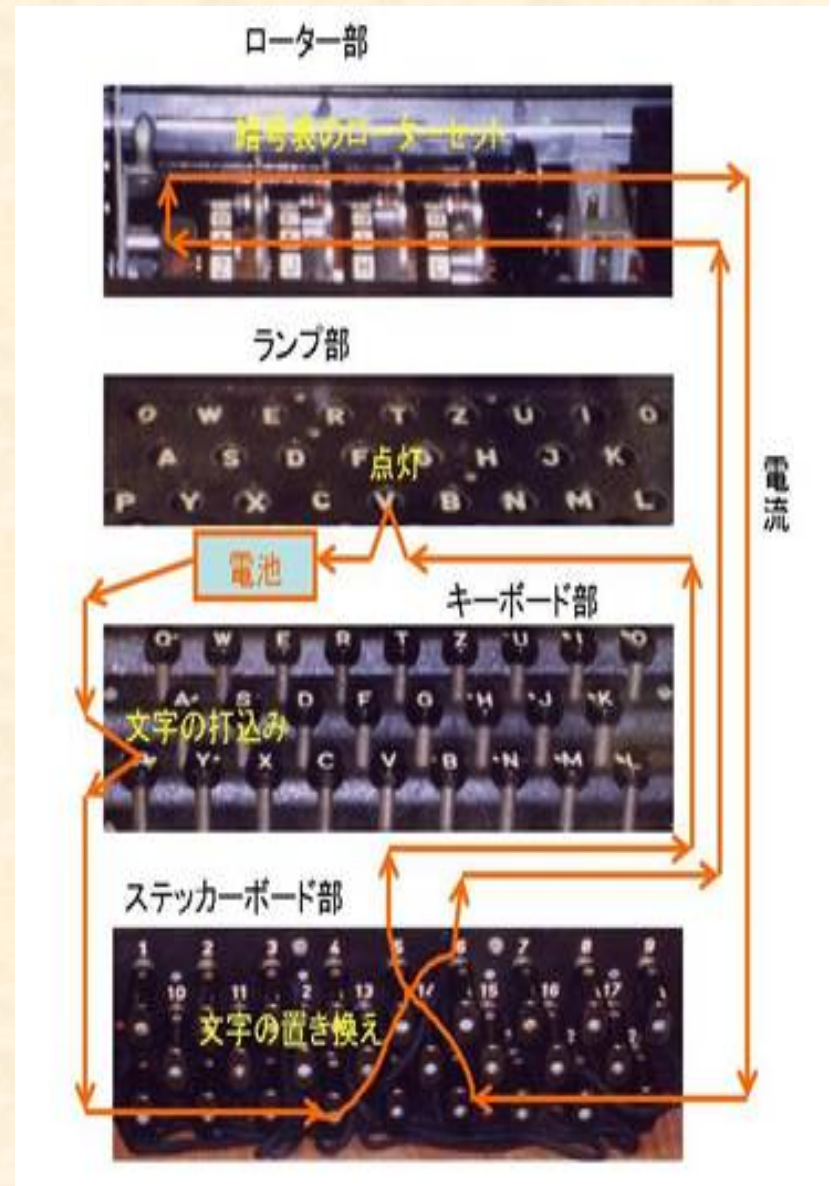
平文 (ひらぶん) plaintext
暗号文 ciphertext

- 暗号と復号は同じ機械でできる。
- 平文の1文字毎に交換関係が変化し、単純な頻度分析 (ポオ「黄金虫」やコナンドイル「踊る人形」) では解読できない。

a m h e r e a b e s l a n e y

- 共通鍵暗号 (送り手と受け手が同じ暗号表を持つ) なので、暗号表を予め別の手段で配らないといけない。

暗号機の内部結線



ローター

- 両面の接点間をランダムな結線で見つけている。
- 結線はローター毎に異なる。固定。
- ローターには26文字が表示されたリングがあり、円周方向に回転させ固定できる。
- ローターが1回転する間に、爪が隣のローターに力を伝え1文字だけ回転させる。歯車計算器の桁上げ機構に類似。

ドイツ軍同士の間通信 (送り手)

暗号表による日毎の初期設定; 日鍵

月毎に配布される

暗号表 (X月Y日)

ロータ を 位置 へ挿入
245 BCD
リング設定

端子 と 端子 を接続
C K
A M
..... 合計10ペア
R W



ロータ 1 2 3 4 5

ロータ 2 4
5 挿入位置 A B C
D
リング設定
スッテカーボード端子
A B C D
K L M R S T
U V W X Y Z

エニグマ暗号機

日鍵設定の可能な総数

- ・ ローター5枚から3枚を選び順に挿入。 ${}_5P_3=60$
- ・ ローターの初期回転位置。 $26 \times 26 \times 26 = 17576$
- ・ ステッカー結線: 26端子を10本のケーブルで。
 ${}_{26}C_2 \cdot {}_{24}C_2 \cdot {}_{22}C_2 \cdots {}_8C_2 / 10! \doteq 1.5 \times 10^{14}$
- ・ エニグマ暗号変換の総組み合わせ数
 $\doteq 1.58 \times 10^{20}$

ドイツ軍同士の通信(送り手) 続き

日鍵は暗号表を盗まれると無効 →

その通信だけに有効な鍵(メッセージ鍵)を送る。

- **インディケータ設定**;ローターを任意に選んだ初期位置まで回転させる。たとえば文字 XATが見える位置に。
- ローター選択、リング設定、ステッカー結線、インディケータ設定を、映画「エニグマ」で見ると
- **インディケータ**;任意に選んだ三つの文字(BGZとしよう)を2回(BGZ BGZ)タイプする。BGZBGZは暗号化されて、**TNUFDQ**(ランプが点灯)に。それを記録する。2回タイプするのは暗号を破られやすくなるので、タイプは1回になった。
- ローターをBGZに合わせ、本番のテキストをタイプする。ランプを読みとり**暗号文**を記録する。

無線送信

- コールサイン、時刻、文字数、暗号機の型を送る。
- インディケータ設定 **XAT**を送る。次に暗号化されたインディケータ**TNUFDQ**を送る。
- 暗号化された本文を送る。

受け手(ドイツ軍)

- 暗号表の日鍵に合わせて、暗号機を設定し、電文の到着を待つ。
- 受信したインディケータ設定 **XAT**にローターを合わせる。
- 受信したインディケータ**TNUFDQ**をタイプするとBGZ BGZという復号されたインディケータが現れるので、ローターをBGZに合わせる。
- 受信した**暗号化本文**をタイプすると、最終的な平文がえられる。

電文を傍受したイギリス軍は、

- 日鍵が不明なので、解読できない。日鍵は暗号表を入手すれば解る。
- 暗号表はスパイ活動または戦場の作戦で入手できた。
- しかし作戦は容易ではないし、入手しても1ヶ月しか有効ではない。
- 暗号機が改造されるとお手上げ。1942年2月、4枚ローターの暗号機がUボートに配置され、暗号は10ヶ月間解読できなくなった。多くの人命が海の底に沈んだ。

ポーランドの発見(1)

サイモン・シン「暗号解読」より

- 暗号文を収集する。
 - たとえば{E . . . X . . . }、{X . . . O . . . }、{O . . . M . . . }
 - 暗号文の1番目と4番目の文字の間の関係に注目
-
- Eから始まる長さ5のループ; $E \rightarrow X \rightarrow O \rightarrow M \rightarrow H \rightarrow E$
 - Aから始めると15の長さのループ; (省略)
 - Vから始めると長さ1のループ(不動点); $V \rightarrow V$
 - Iから始めると長さ5のループ; $I \rightarrow S \rightarrow K \rightarrow U \rightarrow Q \rightarrow I$
 - この4つのループで26文字すべてを尽くしている。

ポーランドの発見(2)

- ・ ループ構造(ループの数とその長さ)は、ローター変換に固有で、ステッカー結線の影響を受けない。ステッカー結線は2文字を入れ替えるだけ。
- ・ 日鍵のすべての組み合わせ(105456通り)を、レプリカの暗号機で試し、日鍵 vs ループ構造の表を作る。
- ・ ある日の暗号文を集めて、最初の6文字からループ構造を抽出し、この表を逆引きすれば、日鍵が発見できる。
- ・ ループを自動的に発見するリレー式の電気回路(Bomba)を作ったが、…… 時間切れ。ドイツのポーランド侵攻が目前に。このノウハウはフランスとイギリスの諜報部へ渡された。

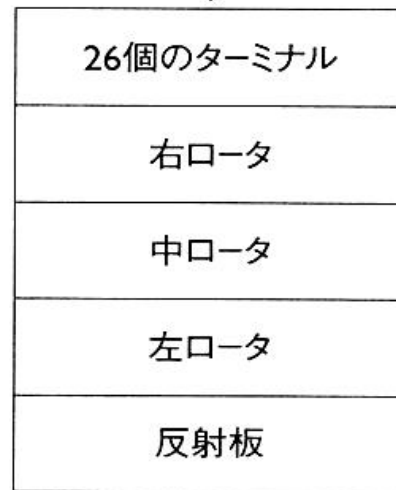
チューリングらによるアプローチ

- ・ 機械化と数理的考察をさらに進め、エニグマ暗号解読の専用マシン； Turing Bombeを設計・製作し、リアルタイムに解読した。
- ・ チャーチルの回顧録にも出てこないほど極秘にされた活動(暗号名 ULTRA)。

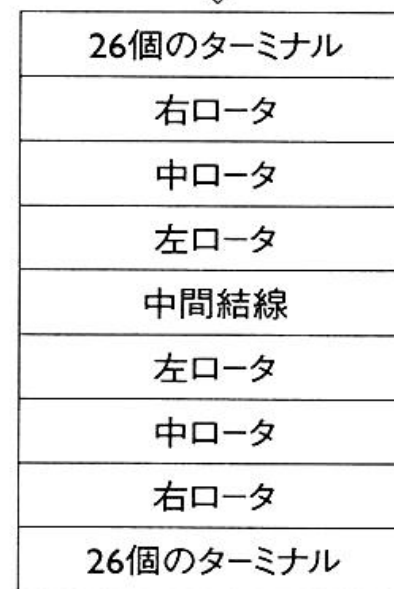
独創性は；

- ・ スクランブラーという3枚のローターの機能をシミュレートする回転ドラムを、多数並列に並べ、暗号機の逐次的な状態を空間方向に展開。
- ・ 電流経路を26本の並行ケーブルとし、26文字を同時並列にスキャン。
- ・ 最も独創的な発明はフィードバック結線、とくに対角結線。
- ・ 数学的な考察(チューリングによる)。

スクランブラー



エニグマ暗号機
電流は上から入り、反射板
で反射し、上から出る。



スクランブラー
上から下へ1方向に電流
が流れる。ポンベは、この
スクランブラーを多数、横
の方向へ並べたもの。

チューリング・ボンベ

- ・ スクランブラーを並列に(文字列の方向)へ並べ、26芯ケーブルで自由に接続できるようにしたもの。

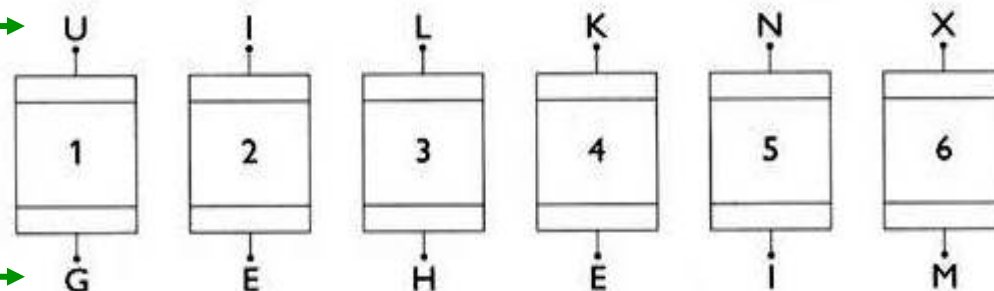
2007年、ブレッチレイにて再現されたチューリング・ボンベは
ブレッチレイパークのHPを見ること。

イギリスの暗号解読の概略

- ・ 諜報活動によって、暗号機の構造、日鍵とメッセージ鍵の存在、通信方法を承知していた。
- ・ 目的;日鍵を1日以内に推定しなければならない。
- ・ まず、クリブ(平文の中に含まれると推定された文字・句)から、文字の関係「メニュー」を取り出し、それに基づいてスクランブラー間を配線する。
- ・ 自動運転;日鍵を順次スクランブラーに設定し、リレー回路で電圧の分布を調べ、正しい(矛盾のない)日鍵とステッカー結線が発見されれば、自動運転をストップする。
- ・ レプリカ暗号機で、ドイツ語として自然な文章に解読されていることを確認。
- ・ 矛盾があるとき、また不自然なとき、次の日鍵を自動的に設定し、自動運転を続ける。

ボンベで正しく解読された状態

- ・ 暗号文「UILKNX」



- ・ 平文「GEHEIM」はクリブ



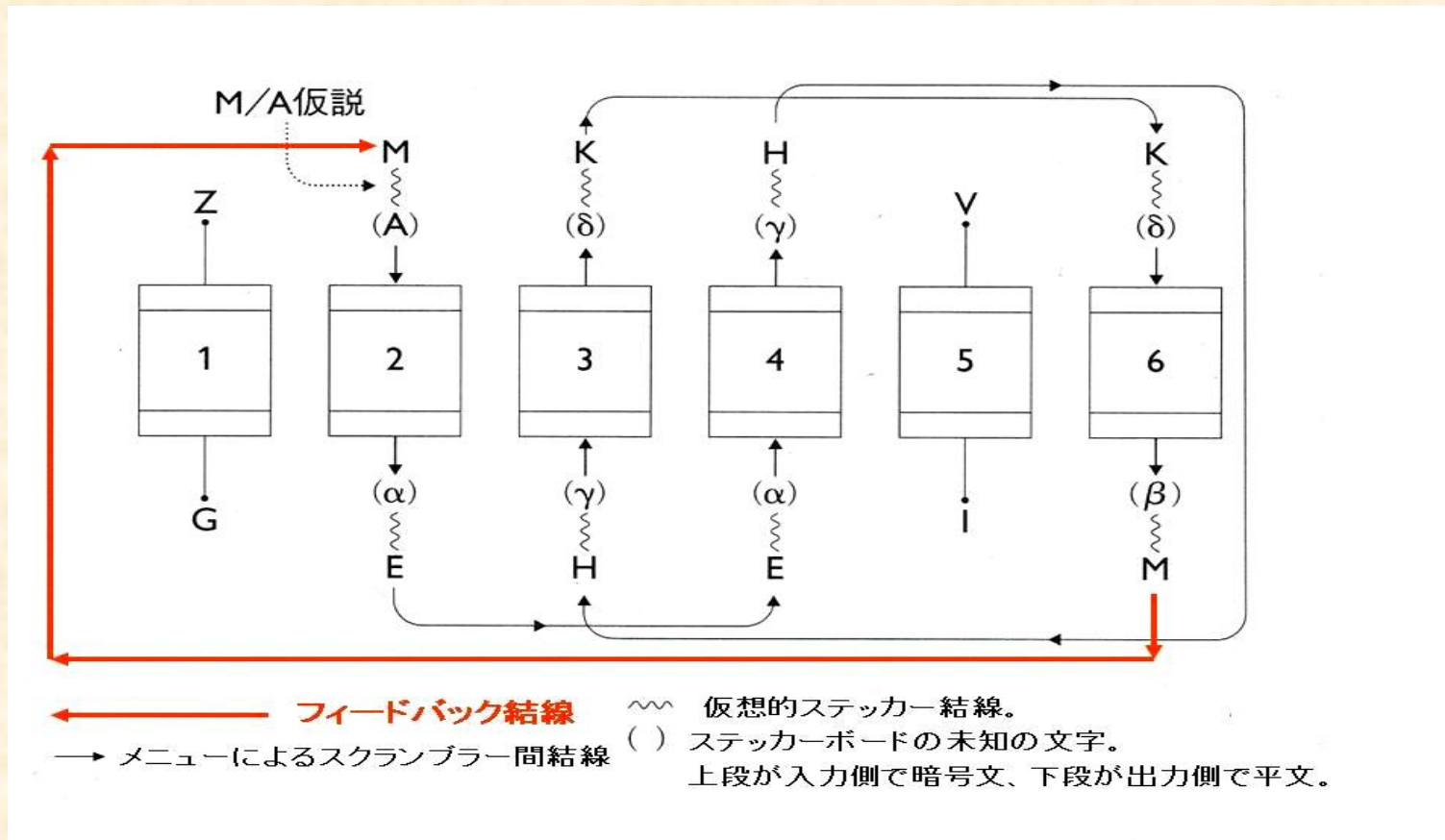
- ・ 電圧のかかっている端子は;

- ・ 暗号側のスクランブラーの U, I, L, K, N, X 端子
- ・ 平文側のスクランブラーの G, E, H, E, I, M 端子

のみで、これら以外の端子には電圧はかかっていない。

- ・ この状態を一気に発見することは、ほとんど不可能。

ステッカー結線を明示すると



暗号文が **Z M K H V K**
 平文の推定(クリブ)が **G E H E I M**
 そこから**M→E→H→K→M**という長さが4のメニューが
 取り出される。メニューを構成する回路全体に電圧は拡がる。

正解の条件

- スクランブラーが正しく設定されている場合;
- (C1) あるX端子に電圧をかけると、そのX端子に限って電圧が現われる。
- (C2) 別のY端子に電圧をかけると、X端子を除いて電圧が広がる。
- (C1)か(C2)であることをリレーで判別し、自動的に停止する。
- クリブが不足してメニューが間違っている場合、「偽の停止」もありうる。
- それを避けるには良質のクリブを獲得する活動が必要。それを実際イアン・フレミングがやっていたという。

正解でない場合

- メニューが適切ではないか、スクランブラーの設定が間違っている場合；
- フィードバック結線を介して、電圧は全部の端子へ広がる。
- これは、述語論理の命題「間違った仮定から導出される定理は全部真になる」の一例であると解釈されている。

ウェルチマンの対角結線

- ・ プラグボード結線は双方向的。
- ・ AとHがつながっていれば、HとAはつながっている(当日は不変)。
- ・ 暗号文に関係なく、予め結線しておいてはどうか？
- ・ それが対角結線。エニグマ暗号全体に有効な計算原理。
- ・ クリブの不足による解読失敗を、大幅に減らした。

暗号解読(キーの探索)時間

- ・ 暗号キーは逐次探索せざるを得ない。ローターの当日の設定総数 $60 \times 17576 = 1054560$ ほぼ100万キー
- ・ 1つのキーの処理速度;リレーとローターの回転速度 = 0.1秒程度。回路の時定数(マイクロ秒)は無視できる。
- ・ 全探索空間 1.58×10^{20} のうちプラグボード結線の空間 1.5×10^{14} は探索する必要はない。
- ・ 全探索時間の期待値は $100万 \times 0.1秒 = 10万秒$ の半分程度 = 15時間
 < ドイツ軍潜水艦が集結するまで = 約1昼夜
- ・ さらに必要なら、複数台のボンベを並列投入した。

終戦

- 1945年5月、ドイツの降伏。突然、ドイツ軍の無線は途絶え、仕事がなくなった。
- ブッレチレイ・パークを去る同僚に向かって、チューリングは未来のコンピュータについて熱く語った。

『数学者はこれからも失業することはない』

ボンベの本質： 非チューリング・マシン

- チューリング・マシンでは、恒等式 $A \equiv B$ は、計算不能。
 - 恒等式を計算するには、ソフトウェアで、代入文を適時くり返す。
 - または、予めプログラミング前に、数学的・物理的考察により変数を消去しておく。
- しかし、ボンベの回路では…… 変数間の等式関係を単純な結線で表現し、計算した！
- アナログ的結線 = 双方向的 並列計算 高速

VS

デジタルマシン = 単方向的 逐次計算 低速

でも、ボンベはチューリング計算可能ではないのか？



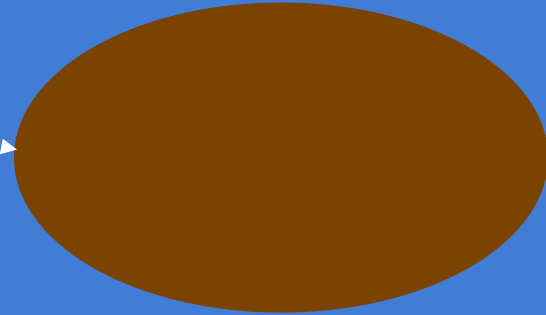
- 数学者からの異議：パソコンで計算できたのだから、チューリング計算可能である。

- 2000年頃のパソコン（もちろんチューリング・マシン）で約3時間を要した。ボンベの所用時間とほぼ同じ。論理素子の速度比は数億倍もある。

パソコンによるシミュレーションは、電圧を、1対1に結線されている端子へ波及させ、それを順次、端子群に変化がなくなるまで繰り返えせば、現実のボンベの物理的状态に収束するという人間による考察に基づいている。

このパソコンは、電圧波及（恒等）関係を、逐次的にシミュレートしているだけ。新しいマシンの創造への指導原理には

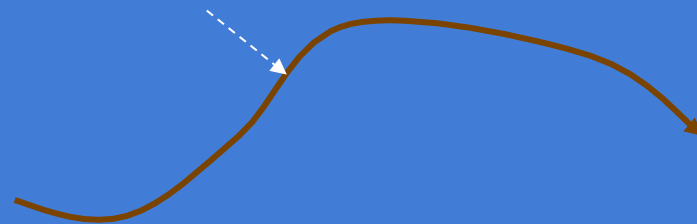
恒等回路の一般化

- 自然法則は恒等式。例；運動方程式。制約プログラミングなど。
- 自由な変数が張る空間  を想定し、ルールや法則でその空間を制約したものが解空間  である。
- そのとき、制約を満たさない矛盾した解の張る部分空間  を崩壊させる（矛盾を掻き出す）強力な手法があれば望ましい。

位相空間 (10^{14})

矛盾した解空間 (10^{14})

法則で制約された解空間 (10^6)



現代暗号も恒等回路で解けるか？

- どんな暗号でも成り立つ恒等式がある。それは

国家秘密

矛盾を検出する技術
的方法は？

しかし.....残念ながら、
鍵が逐次的になる。空
間的に展開して並列処
理すればいいが、配
線・ピンリミットにな
るだろう。

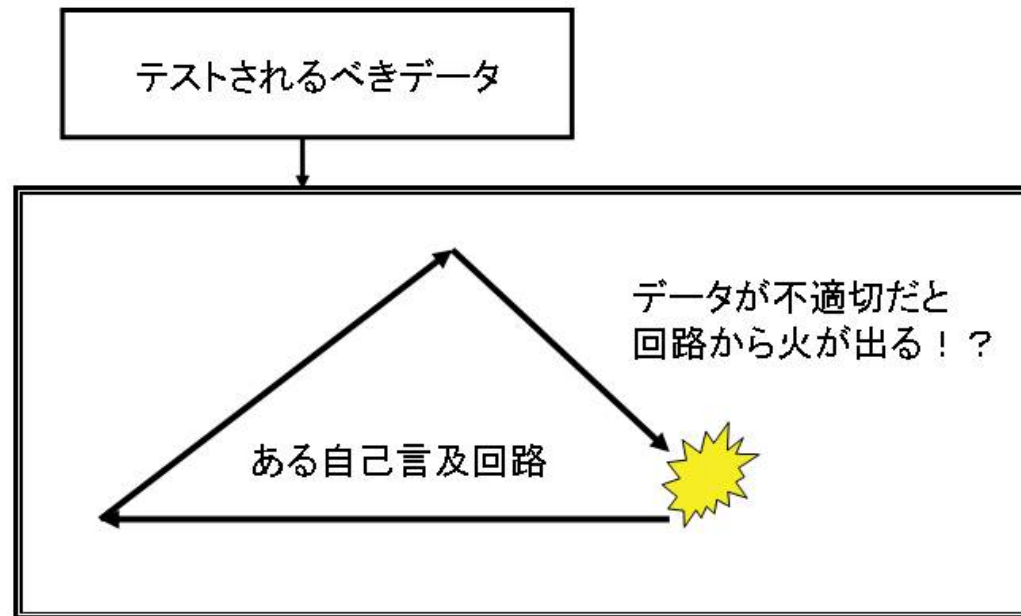
“SUPERSCRITCHER” (scratching out contradictions) . The sensitive
circuitry of CRAY computers hides some of the cryptanalytic algorithms.

128ビットDES暗号を並列処理するときの推定

- 並列処理速度 10^{14} infer/sec
(回路時定数 = 10^{-8} sec, 並列度 10^6 チップ と仮定)
- 逐次処理量 $2^{(128-n)}$, n はクリブによる限定
- 解読を1時間以内にやるには、

$$2^{(128-n)} \times 10^{-14} < 3600 \quad n > 63$$

越中コンピュータで チューリングを越えたい！



チューリング・マシンの可能性

自己参照・自己改変

を制限した20世紀

チューリングのACEやフォン・ノイマンタイプのマシン(EDSAC, EDVAC)では、自分で自分（プログラム）を変える能力があった。

- しかし、1950年代、Bレジスターによるプログラム修飾とメモリ領域保護によってその牙を失った。
- その後、自己書き換え機能は、セキュリティ関連や進化計算などでは生き残っている。
- ハードウェアで自己改変（FPGAなど）すると暴走し、ゴミをまき散らす。このとき、恒等回路でゴミのスクラッチをやれないか？

おわりに

チューリングの計算パラダイムは、歴史の産物。
強力だが、低速。
パラダイムを乗り越えないと、
高速化はできない

参考資料

星野 力「甦るチューリング」NTT出版 2002

星野 力「チューリングを受け継ぐ」頸草書房 2009

チューリングの主要な論文と評論は、

B. Jack Copeland (ed.) “*The Essential Turing, the ideas that gave birth to the computer age*” Oxford Press (2004)

Christof Teuscher(ed.) “*Alan Turing: Life and Legacy of a Great Thinker*” Springer (2004)

FIN

Background Color
R 206 G 185 B 142

戦後、計算可能性の学者へ変身

- 1945年（33歳） 国立物理学研究所に勤務。チューリング・マシンを実現しようとした。
- 1946年（34歳） ACEコンピュータの提案。しかし完成は遅れ、いくつかのアイデアは未完で終わった。
- 1947年（35歳） プログラミング、人工脳（人工知能）の議論。チューリング・テストの提唱。

1940年前後のコンピュータ開発

- 機械式や電気式のアナログ・シミュレータ（微分解析機、潮汐解析、交流計算盤）や、デジタル計算機は、1938-41年 Zuse Z1,Z2,Z3 電気機械式。プログラム制御
- 以下は真空管回路によるデジタルマシンに限定；
- 1939年 ABC 一次連立方程式求解機
- 1943-44年 COLOSSUS 暗号解読。プログラム制御
- 1946年 ENIAC 微分方程式解析機。1948年 改造ENIAC プログラム制御
- 1948年 Baby Mark-I 最初のコンピュータ（可変プログラム内蔵、TM)
- 1949年 EDSAC ソフトウェアを備えた、最初の使えるコンピュータ。
- 1950年 Pilot ACE チューリングのACEのプロトタイプ

チューリングとフォン・ノイマン(1)

フォン・ノイマンが単独の著者名で配布した「EDVACに関する報告書--草稿」(1945)

- プログラム可変内蔵方式（プログラムが高速メモリに内蔵され、データと区別なく演算処理される方式）の構想を最初に公表した論文
- ノイマンがコンピュータの基本概念であるプログラム内蔵方式の発明者であると一般には受け取られている。
- そこではプログラムとデータを区別しないことの理由として、それが「望ましい」と書かれているだけで、積極的な理由は示されていない。単なるハードウェアの節約のようにも読める。

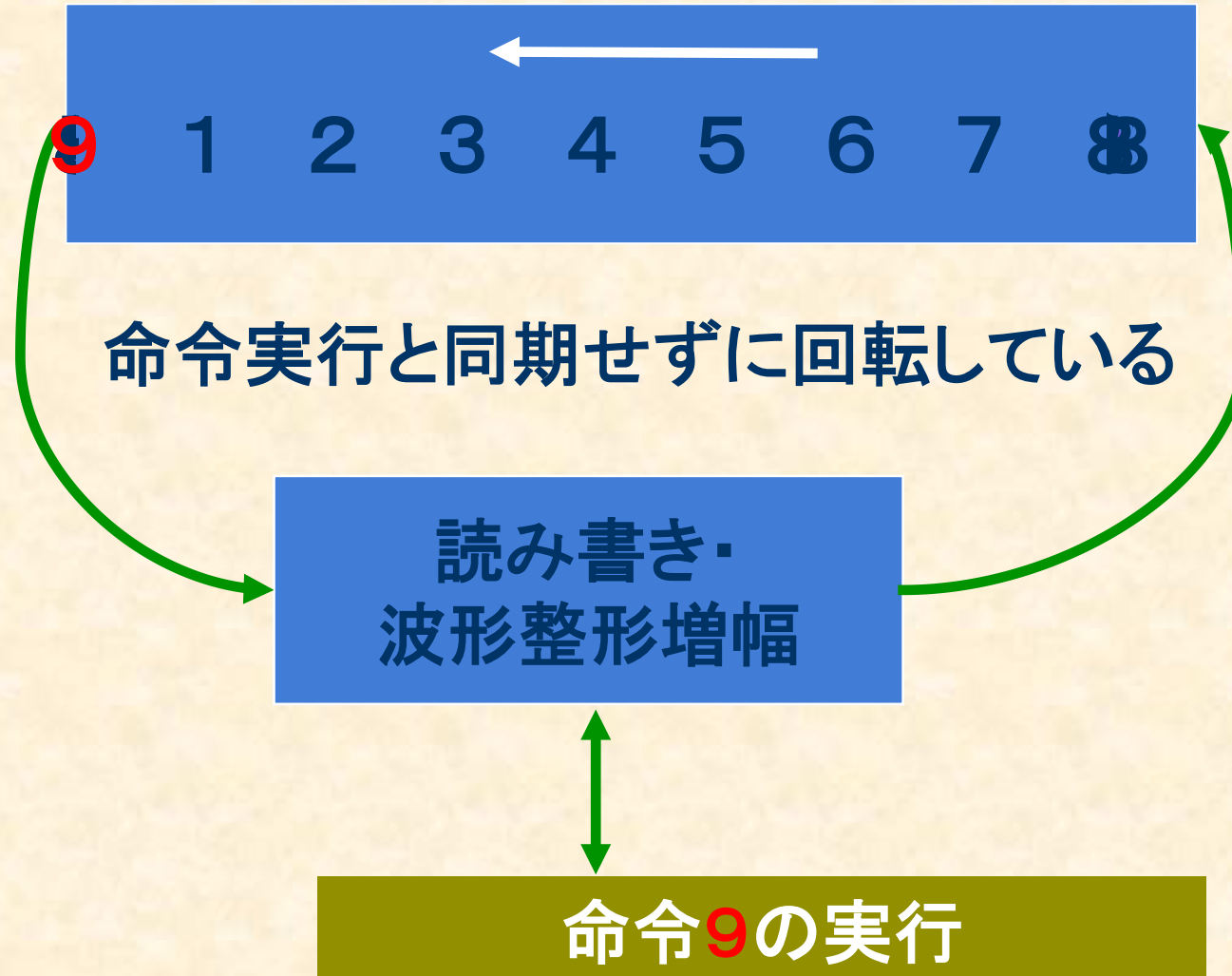
チューリングとフォン・ノイマン(2)

- これに対してチューリングは、積極的に万能チューリングマシンUTMを作ろうとしていた。彼は人工脳を作ること、今の言葉では人工知能のためにACEコンピュータを作ろうとしていた。
- UTMでは本質的にプログラムとデータを区別しない。1947年のロンドン数学会における講演でチューリングは、ACEとUTMとを明確な言葉で関係づけている。
- UTMはコンピュータ・アーキテクチャとしては、プログラム可変内蔵方式に他ならない。
- 疑いもなく、世界で最初にプログラム可変内蔵方式の基礎理論を提唱し、その方式を論じたのはチューリングである。

チューリングとフォン・ノイマン(3)

- ノイマンは、チューリングがプリンストン大学に留学していた1938年頃から、「計算可能数」の論文とUTMについて知っていた。
- しかし、EDVACコンピュータと計算可能数論文との関係に気がつくのは、1945年頃と想像される。
- EDVACの議論に加わっていたアーサー・バークス（後にミシガン大学教授）が、「エッカートらが（読み取り専用の）高速メモリをEDVACに使うことを議論しだした後で、ノイマンは初めてそれを読み書き可能なプログラムメモリにすること、プログラム可変内蔵方式の議論を始めた」と証言している。
- ノイマンは、ロスアラモスの研究者たちに「コンピュータの基礎概念はチューリングに負っている」と強調していた。

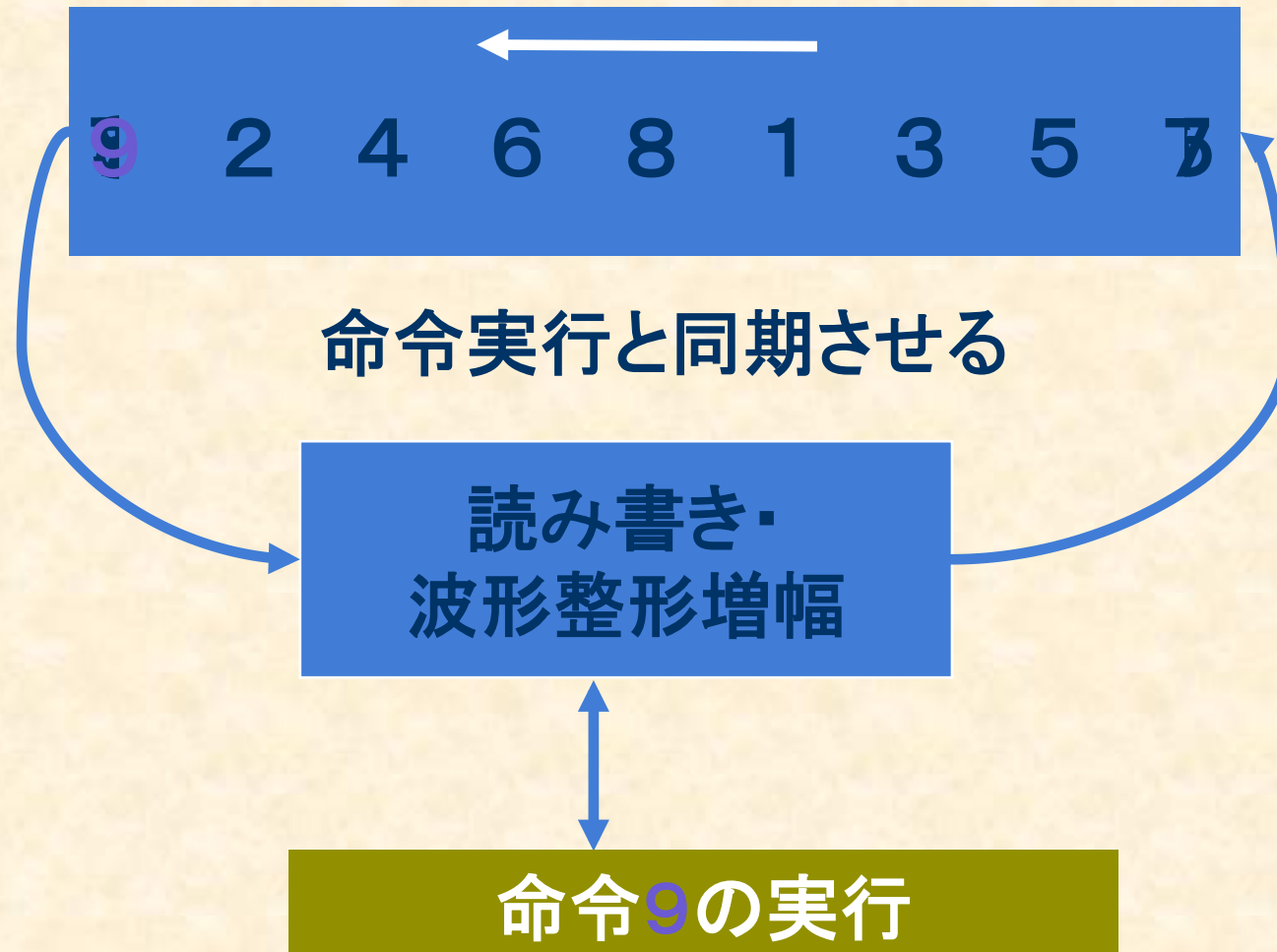
超音波遅延メモリ



ACEコンピュータの特徴(1)

- 最適コーディング。命令実行が終わるとちょうど次の命令がメモリから出てくるよう命令を並べる。メモリ=大きな遅れをもつ超音波遅延線（管）。
- チューリング・マシンの実現。できるだけソフトウェアで計算しようとした。
- 専用ハードを多用するのは「悪しきアメリカ流」。

最適コーディング



ACEコンピュータ、その後

- ノイマン型への妥協を重ね、ACEは1957年完成、数値的な計算に使われた。
- チューリングの元にいたアメリカ人Huskeyは、1956年にBendix G-15を商品化。チューリング的最適化思想を引き継いでいた。
- 日本最初の輸入コンピュータ。国鉄（MARS）や三菱電機（MELCOM）へ大きな影響を与えた。

Bendix G-15

日本最初の輸入コ
ピュータ（195
7）

三菱電機伊丹研究所、
国鉄、日航など

しかし、チューリング
的プログラミングは、
インタコムというソ
フトによって、隠さ
れていた。

マンチェスターへ：計算生命の先駆者

- 1948年（36歳） マンチェスター大学へ赴任

• 世界最初の
コンピューター Baby
Mark-I が
待っていた

チューリングの勤めた
マンチェスター大学 →

住んでいた郊外(Wilmslow) ↓



チューリング・パターンの計算

- チューリングはコンピュータを使う側に廻った。生物の形態形成の計算。
- 1990年代になって、やっと生物分野で認識された（理研・近藤滋氏：タテジマ・キンチャクダイ）。
- われわれは容易にチューリングの掌から出られない？

反応拡散方程式

$$\frac{\partial U}{\partial t} = D_u \nabla^2 U - UV^2 + F(1 - U)$$

$$\frac{\partial V}{\partial t} = D_v \nabla^2 V + UV^2 - (F + k)V$$

謎の自殺

- **1952年（40歳）** ホモセクシャルとして有罪。
- **1954年（42歳）** 自殺（6月7日）。

Alan Turing's house
near Wilmslow, Cheshire,
where he took his own life
on 7 June 1954. He took
potassium cyanide.

fin