

# 第4章

ソフトウェアが拡張する情報処理機械

# 目次

- サブルーチン
- ライブラリ
- 命令シミュレーション
- プログラミング言語と翻訳
- 資源の仮想化
- 資源の抽象化

# EDSAC : ソフトウェアの原点

- 命令の呼び名 + 10進数番地の命令を2進数変換し ,  
メモリに配置

EDSAC : イニシャルインプットルーチン ( たった41 命令 )  
システムソフトウェア

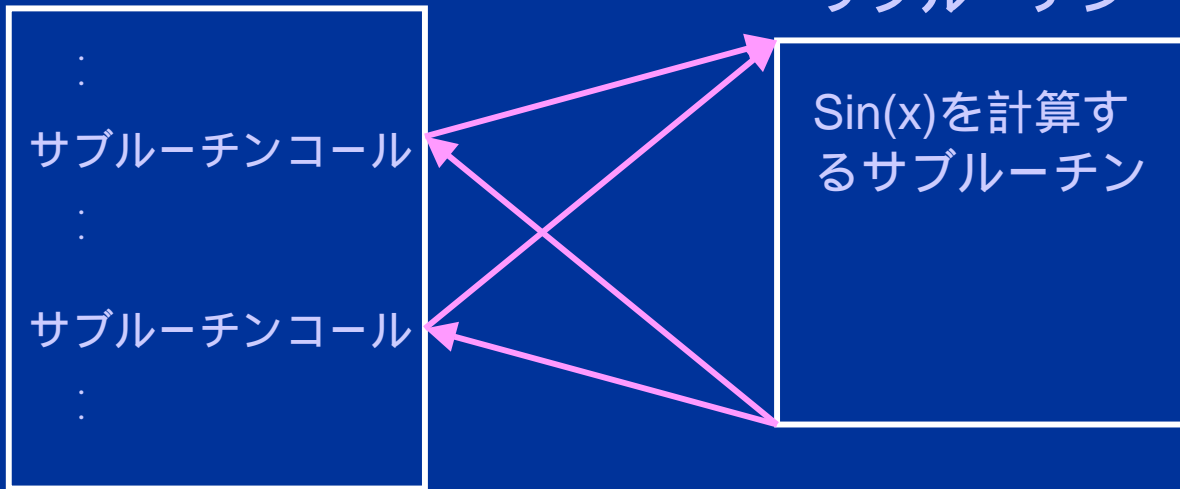
- サブルーチン & ライブラリ
- 命令シミュレーション  
部品化 , 構造化プログラミング

# サブルーチン

EDSAC : 17種類の単純な命令 . 割り算さえなし .

プログラムをすることによって , ハードウェアに備わった命令と同等の機能を追加 .

例 :  $\sin(x)$  を計算するには  
主ルーチン



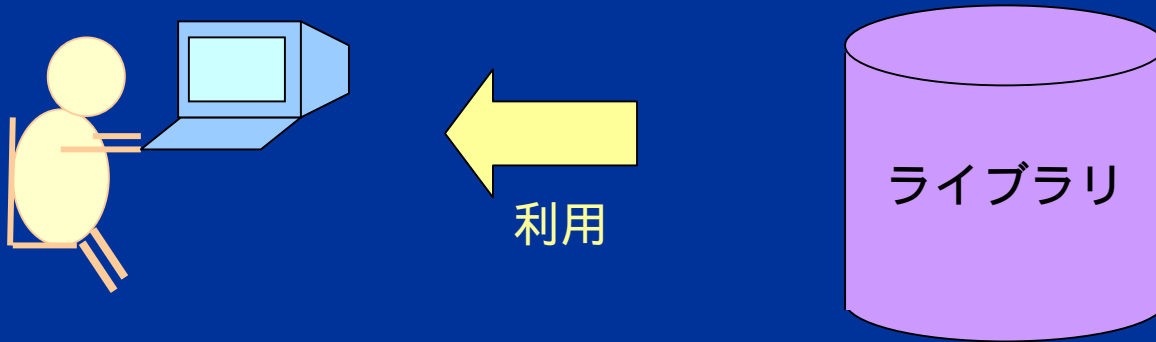
主ルーチン : 主プログラム

サブルーチン : プログラムの部品

主ルーチンも , サブルーチンもメモリーのどこかに置かれる . 再利用する部分を何度も主ルーチンに含むことなく , それらを“呼ぶ”だけでよい .

# ライブラリ

EDSAC：割り算や有用な関数群のサブルーチンを個々にユーザが作らなくても，あらかじめ用意して提供．（サブルーチン）ライブラリ

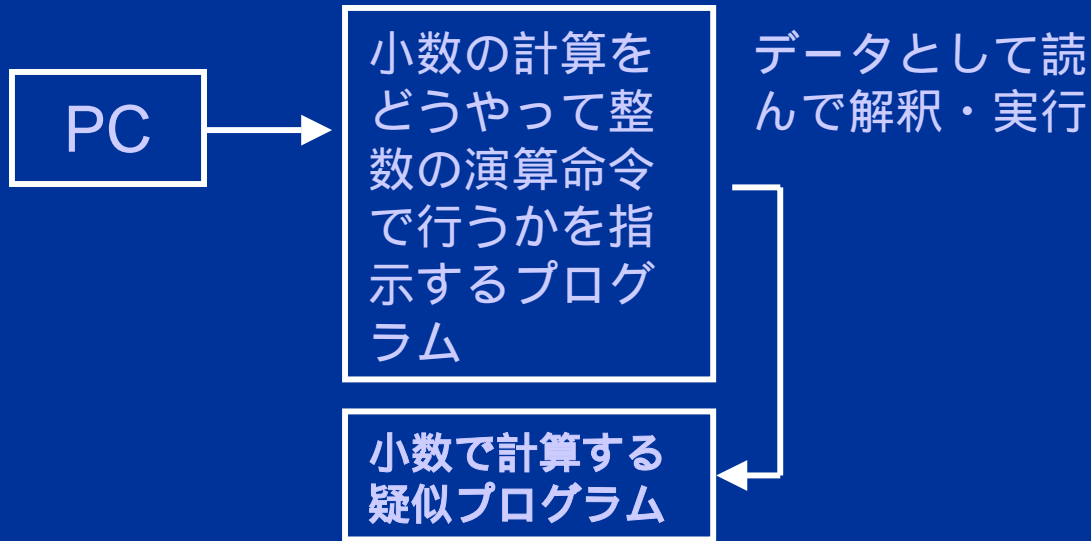


- 金物（hardware）と違って紙テープに記録．読め込めば純粹に情報
- 必要に応じて開発し，変更が容易
- hardware による命令と同等かそれ以上の機能を提供

hardware に対して software と呼ばれるようになった．

# 命令シミュレーション

ハードウェアにない命令  
ビット列を解釈して実行  
する



小数演算，複素数の計算，  
など．

ハードウェアにない命令  
があたかもあるかのごと  
く動作する．

その後：ハードウェアにない命令を実行しようとする時、それを解釈するところへ自動的に飛び、処理して戻る。命令とデータがともにビット列であることを利用。

命令シミュレーションやサブルーチンを呼ぶ感覚さえない。

# ソフトウェアの発展の基盤になったもの

EDSAC以後

プログラミング言語と翻訳  
資源の仮想化，抽象化

# プログラミング言語と翻訳

人間の問題解決の思考レベルに近い言語でプログラミング．機械語に翻訳して実行．

10 個の数値を加えるED9900プログラム．構造化Basic風なら

！これより右はコメント．10個の数値の並びの合計を求めるプログラム

DIM X(10) ! 10個の数値の並びである配列を使うことを宣言．

配列の初期化 ! 要素は1から10まで．

LET total = 0 ! total の値を 0 に初期化

➡ FOR i = 1 TO 10 ! i を 1 から 10 まで繰り返す

LET total = total + X(i) ! total <- X(1) + X(2) + ... + X(10)

NEXT i !

PRINT total ! total を出力

END ! プログラムの終わりには必ず END を付ける．



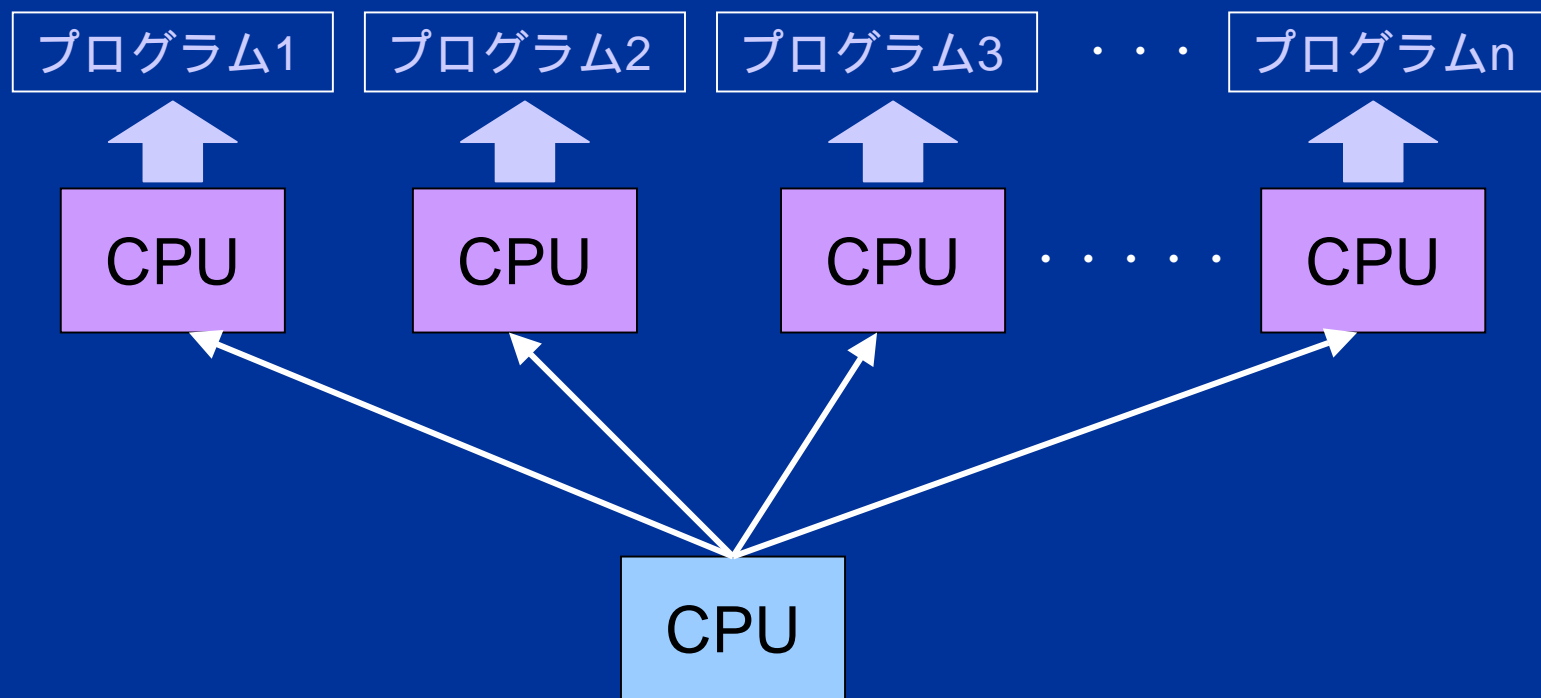
資源の仮想化, 抽象化

# CPUの仮想化

CPUが一つしかなくても、同時に複数のプログラムを実行させたい

- 例：
- ・ 複数ユーザのプログラムを一台の大型コンピュータで処理
  - ・ 一人のユーザでも、ブラウザを見ながら、ワープロソフトや表計算ソフトを使う。

CPUを必要なだけあるように見せたい。



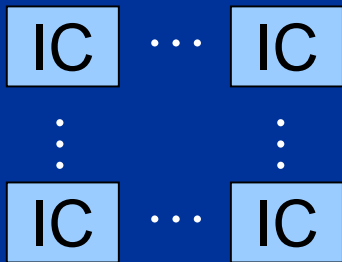
# メモリーの仮想化

高速だが高価な主メモリー

低速でも安価な二次メモリー

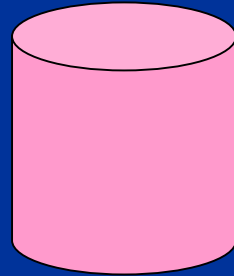
<半導体メモリーとハードディスク>

高速で大きい主メモリーがあるかのように見せる。

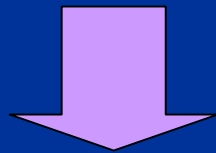


主メモリー

+

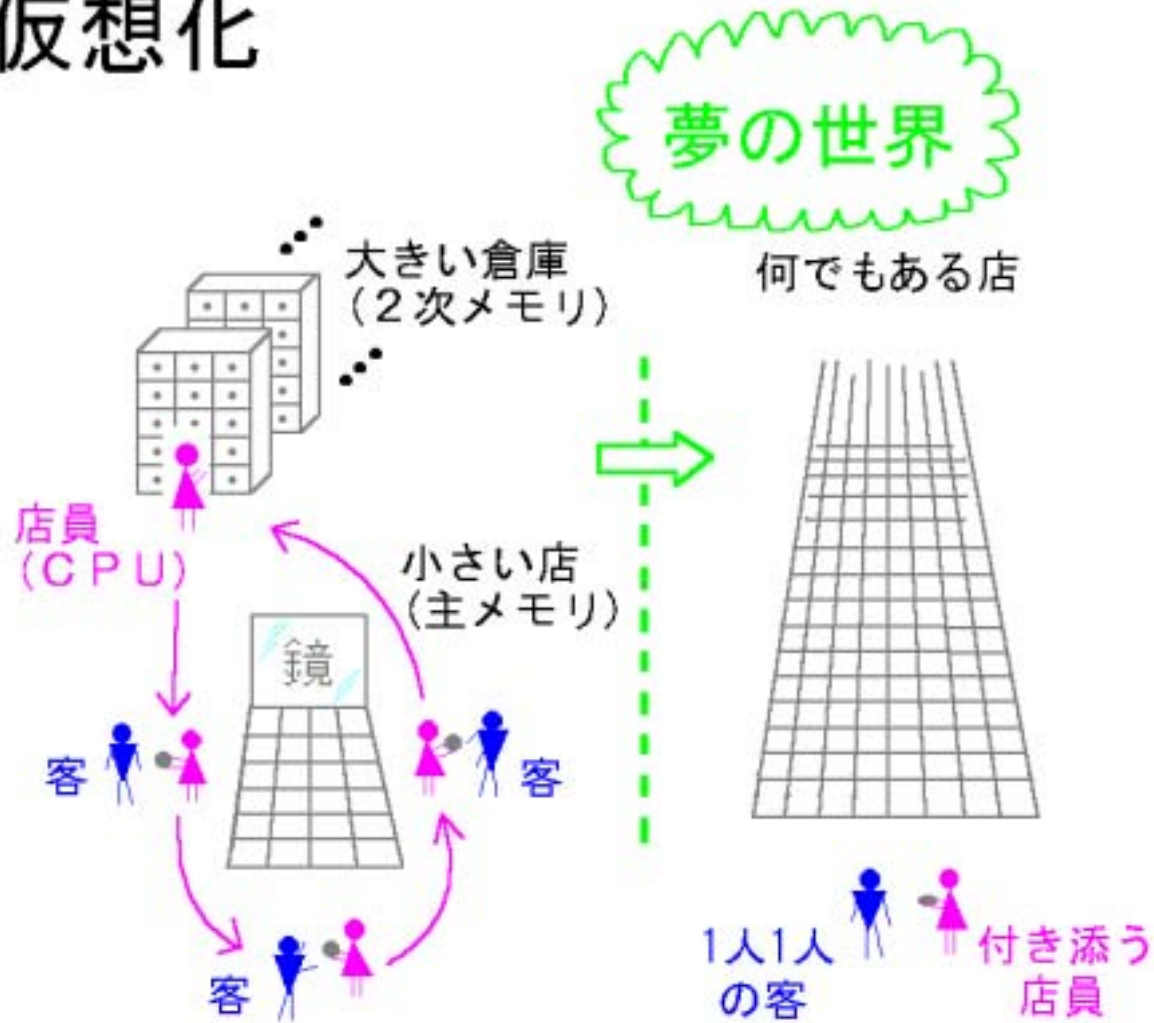


二次メモリー



仮想メモリー

# 仮想化

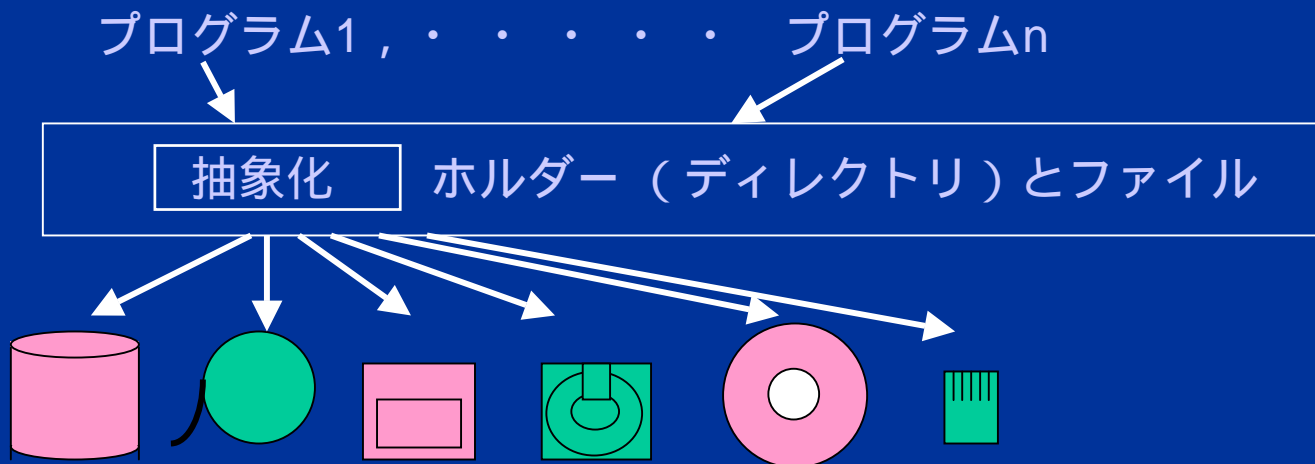


## 資源の仮想化

# 抽象化

一群の装置を，機種の違いや仕様の違いを隠蔽して，ある抽象的な概念で利用できるようにする．

## プログラムからの利用



2次記憶装置：ハードディスク，磁気テープ，フロッピーディスク，MOディスク，DVD，メモリーカード，. . .

一つのプログラムがどれかの装置を使い始めたら，使い終わるまで他を排除 = 排他制御．

# ソフトウェアが拡張する情報処理機械

問題ごとの対話言語

アプリケーションが提供する、  
さらに人間の活動よりのコンピュータ

プログラミング言語

オペレーティングシステムが  
見せる仮想のコンピュータ

機械語

ハードウェアの  
提供する命令

# まとめ

コンピュータシステム：数十万～数千万行のプログラムからなる巨大なシステム

その機能拡張のメカニズムの原点にプログラム内蔵方式  
ループのたびに少しずつ変わる命令を容易に実現  
それがなくても済む機能の発明 使われなくなった  
プログラムを処理するプログラム  
(アセンブラ, コンパイラなど)の発想を容易に  
仮想化, 抽象化の思想を形成

× プログラムを“悪さ”するプログラムに変える = ウィルス

プログラム内蔵方式こそ, 現在のコンピュータとソフトウェアの発展を作った最大の概念