

コンピュータアーキテクチャ

白井暁彦 担当分

2012/4/17更新分

第1回のおさらい

- * コンピュータアーキテクチャとはなんだろう？
 - * 「自分に関係がある」とわかればよろしい
 - * 基本情報技術者試験のハードウェア関連に役立つ
 - * ゲーム機や新しいプラットフォームの開発者なら必須
- * 関連用語のおさらい
 - * Architecture, access, algorithm, software, hardware, bit, byte, binary, digit, program
 - * {personal, micro-, mini-, office, mainframe} computer, workstation...
 - * OS, SDK, API,...

コンピュータアーキテクチ関連用語

1123158

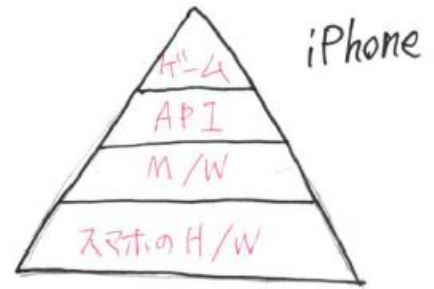
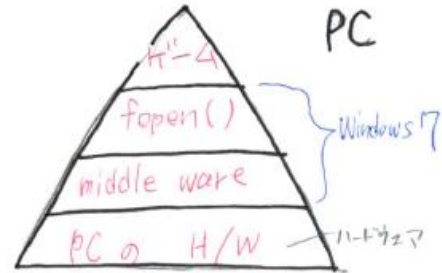
田邊 圭琳

4月10日

今回

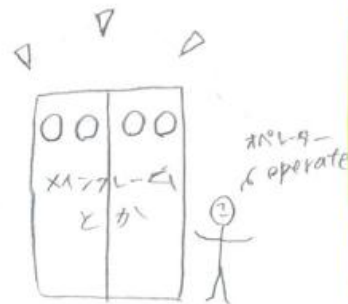
意味を理解できた用語

- アーキテクチ (architecture) ... 建築技術, ハードウェアの基本設計のこと.
建築 技術
- CAD (Computer Aided Design) ... コンピュータを用いた製図システムのこと.
支援 設計 構造
- アルゴリズム (Algorithm) ... 問題を解くための効率的手順を表現した物
- bit (binary digit) ... デジタルコンピュータにおけるデータの最小単位
Binary No. は、2進法 2進法



コンピュータ種類

- パソコン (personal computer) ... 個人用。
- マイコン (micro computer) ... 小
- ミニエ (mini computer) ... 大
- オフコン (office computer) ... 大
- ワークステーション (work station)
- Xインフレーム (main frame) ... OSなし。



Operating System ... Software ↔ Hardware

● API (Application Program Interface)

OSやソフトウェア向けの Software を開発お際に使用する命令と関数の集合のこと。

VE: スカラー量
ベクトル演算
Vector Unit



第1回 追加分

* 「ハミング距離」 Hamming Distance

- * 等しい文字数を持つ二つの文字列の中で、対応する位置にある異なった文字の個数(Wikipediaより)
- * 2つの2進数があり、お互いの隔たりを表すために桁ごとに排他的論理和した値の総計をさす(教科書 p.2 より)
- * 例:
 - * 10**11**01 と 10**01**01 の間のハミング距離は 2 である。
 - * 2**14**3**8**96 と 2**23**3**7**96 の間のハミング距離は 3 である。
 - * "**t**oned" と "**r**oses" の間のハミング距離は 3 である。

[bitwise eXclusive OR]

1011101

1001001

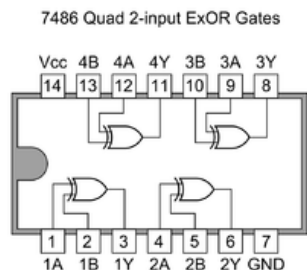
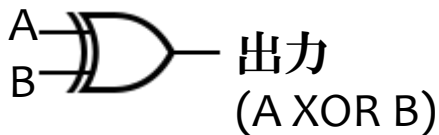
----- (XOR)

0010100 => 2

[真理値表]

| 入力 A B | | 出力 A XOR B |
|-----------|---|---------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

* XORゲート



第1回 追加分

* 「パリティ検査」 parity check

- * ある2進数内における1の個数が偶数か奇数かを手がかりに行う符号誤りの検査方法である(教科書より)
- * ある数字の並び(大体2進数)の合計が偶数・奇数かどうかを比較する事により、**通信の誤りを検出する技術**。
偶数を **even parity**、奇数を **odd parity** という。

* その誤りを調べることをパリティ・チェック (parity check) という (Wikipedia)

* 多くの場合oddパリティが用いられる。

| 7ビットのデータ | パリティビット付きのバイト | |
|----------|---------------|----------|
| | even | odd |
| 0000000 | 00000000 | 10000000 |
| 1010001 | 11010001 | 01010001 |
| 1101001 | 01101001 | 11101001 |
| 1111111 | 11111111 | 01111111 |

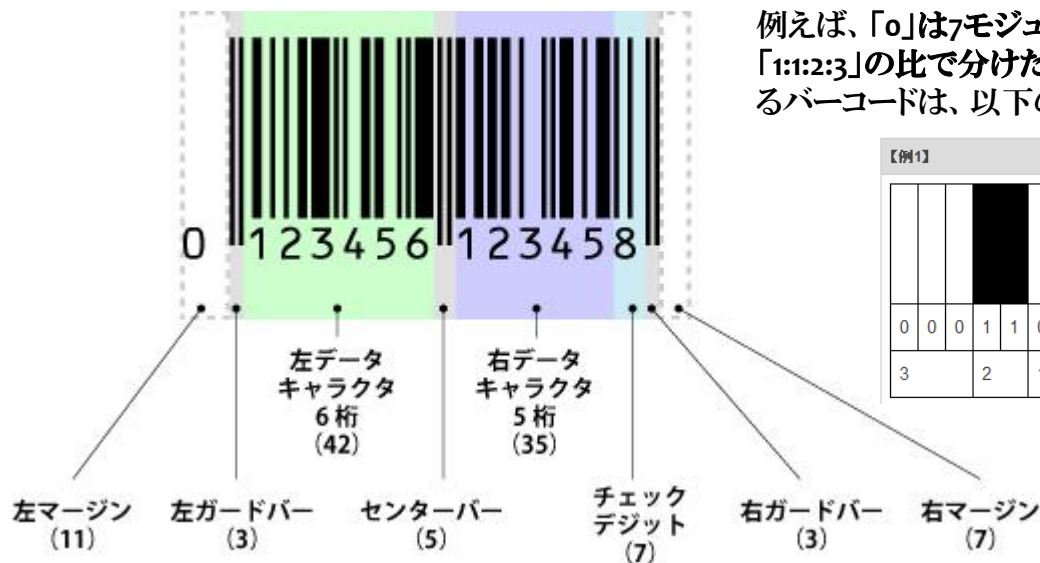
第1回 追加分

- * 「パリティ検査」 parity check
→バーコードのしくみ

* <http://piyajk.com/archives/321> より

- * 国コード (2桁) + メーカーコード (5桁 or 7桁) + 商品コード (5桁 or 3桁) + チェックデジット (1桁) ... の13桁.

例えば、「0」は7モジュール分の面積を「3:2:1:1」またはその逆並びの「1:1:2:3」の比で分けたもの、と決められています。この条件で作成できるバーコードは、以下の4つが考えられます。



| 【例1】 | 【例2】 | 【例3】 | 【例4】 |
|---------------|---------------|---------------|---------------|
| | | | |
| 0 0 0 1 1 0 1 | 1 0 1 1 0 0 0 | 0 1 0 0 1 1 1 | 1 1 1 0 0 1 0 |
| 3 2 1 1 | 1 1 2 3 | 1 1 2 3 | 3 2 1 1 |

クロが奇数個を「奇数パリティ」
クロが偶数個を「偶数パリティ」

第1回 追加分

* パリティとチェックディジット

各数字に対応する7モジュール分の塗り分け一覧

| 数字 | 面積比 | 左データ用 | (1桁目用) | 右データ用 |
|----|--------------------|--------|--------|-------|
| | | 奇数パリティ | 偶数パリティ | |
| 0 | 3:2:1:1 or 1:1:2:3 | 000 | 0 | 000 |
| 1 | 2:2:2:1 or 1:2:2:2 | 00 | 0 | 00 |
| 2 | 2:1:2:2 or 2:2:1:2 | 00 | 00 | 00 |
| 3 | 1:4:1:1 or 1:1:4:1 | 0 | 0 | 00 |
| 4 | 1:1:3:2 or 2:3:1:1 | 0 | 00 | 00 |
| 5 | 1:2:3:1 or 1:3:2:1 | 0 | 0 | 00 |
| 6 | 1:1:1:4 or 4:1:1:1 | 0 | 0000 | 0000 |
| 7 | 1:3:1:2 or 2:1:3:1 | 0 | 00 | 00 |
| 8 | 1:2:1:3 or 3:1:2:1 | 0 | 000 | 000 |
| 9 | 3:1:1:2 or 2:1:1:3 | 000 | 00 | 000 |

1桁目の表し方

| 数字 | 偶奇の組み合わせ方 |
|----|-------------|
| 0 | 奇 奇 奇 奇 奇 奇 |
| 1 | 奇 奇 偶 奇 偶 偶 |
| 2 | 奇 奇 偶 偶 奇 偶 |
| 3 | 奇 奇 偶 偶 偶 奇 |
| 4 | 奇 偶 奇 奇 偶 偶 |
| 5 | 奇 偶 偶 奇 奇 偶 |
| 6 | 奇 偶 偶 偶 奇 奇 |
| 7 | 奇 偶 奇 偶 奇 偶 |
| 8 | 奇 偶 奇 偶 偶 奇 |
| 9 | 奇 偶 偶 奇 偶 奇 |

郵便カスタマバーコード



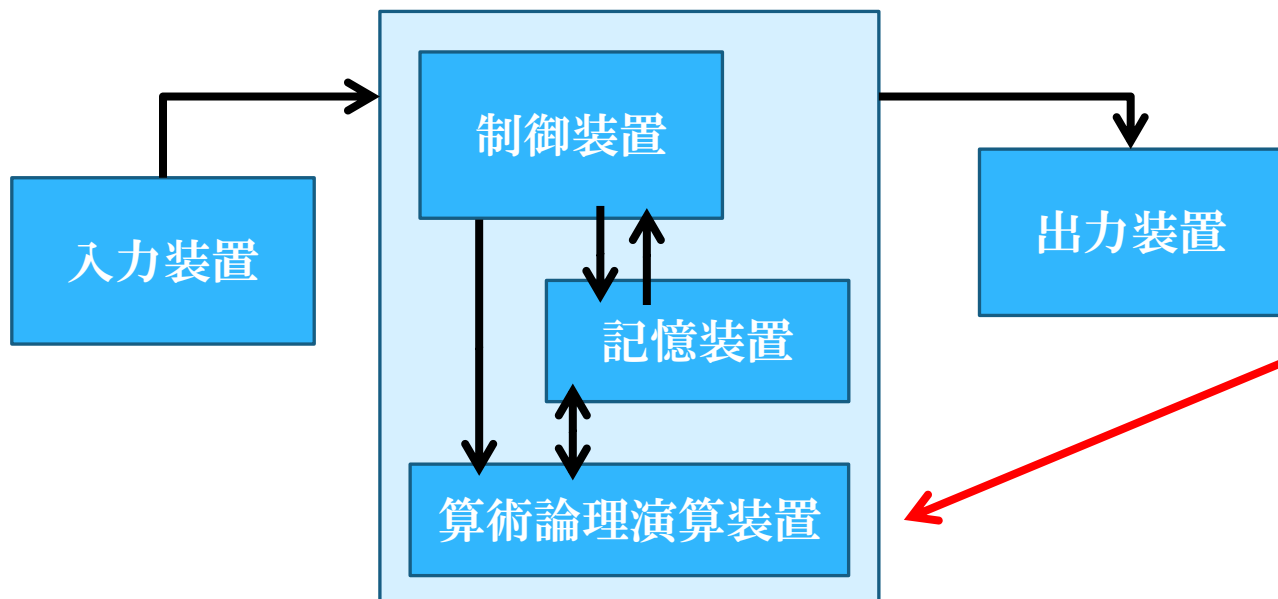
| | | | |
|--------------------------------|-------------------------------|---------------------------|-------------------------------------|
| JAN8 F01234666 | JAN13 10000001234666 | Code39 123466 | Code39Extended 123466 |
| Code_2_5_interleaved 123466 | Code_2_5_industrial 123466 | Code_2_5_matrix 123466 | Code128A 123466 |
| Code128B 123466 | Code128C 123466 | Code93 123466 | Code93Extended 123466 |
| CodeMSI 123466 | CodePostNet 123466 | Codabar 123466 | EAN8 0123 4666 |
| EAN13 0 000001 234666 | CodeEAN128A 12346 | CodeEAN128B 12346 | CodeEAN128C (UCC/EAN-128) 123466 |
| UPC_A 12346678910 | UPC_E0 123466 | UPC_E1 123466 | UPC_Supp2 12 |
| UPC_Supp5 12346 | | | |

第1回 追加分

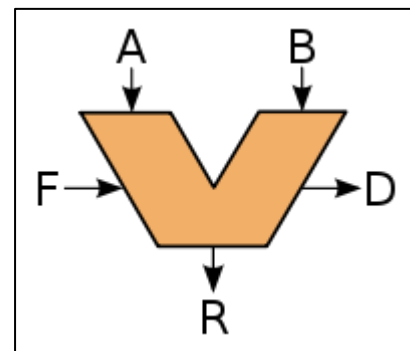
- * ノイマン式コンピュータ von Neumann-type computer
 - * ノイマンは、プログラムをハードウェアから独立させてデータとして外部から与え、汎用のハードウェアでこれを実行させる方式を発表した
 - * これがノイマン型コンピュータである。ソフトウェア(プログラム)という概念の誕生もこのときであった。
 - * ノイマン型アーキテクチャの5つのハードウェア
 - * 入力装置:プログラムやデータを入力して,
 - * 記憶装置:命令やデータを記憶して,
 - * 制御装置:プログラムの命令に伴う制御信号を生成して,
 - * 演算装置:必要あればデータの演算を行い,
 - * 出力装置:その結果を出力する

第1回 追加分

* ノイマン式コンピュータ von Neumann-type computer



算術論理演算装置
ALU(Arithmetic Logic Unit)

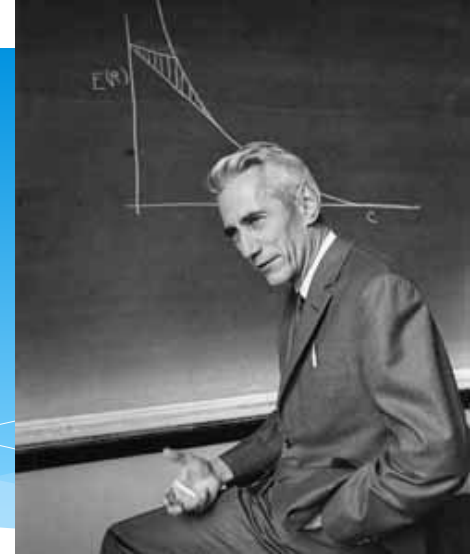


非ノイマン型コンピュータ

たとえば電卓のような汎用ではないクラシックな計算機、
ノイマン型計算機の特徴であるプログラム内蔵方式、
逐次制御方式、線形記憶方式等を一部否定した計算機システム。
量子コンピュータなども非ノイマン型コンピュータといえるかもしれない。

ALUを表す記号。AとBは入力(オペランド)でRが演算結果。Fは制御部からの入力、Dは出力ステータス。

シャノンの情報理論



* クロード・エルウッド・シャノン
Claude Elwood Shannon

* デジタル回路設計の創始者

- * 1937年のマサチューセッツ工科大学での修士論文「継電器とスイッチ回路の記号論的解析」において、電気回路でブール代数を扱うことができることを示した。
- * シャノンはこの論文で、スイッチのオン・オフを記号論理の真・偽に対応させると、スイッチの直列接続はANDに、並列接続はORに対応することを示し、あらゆる論理演算がスイッチ回路で実行できることを証明した。
- * これによって、ただの計算機械(コンピュータ=computer)が、現在のような高速の論理演算機として活躍することが可能となった。
- * ハーバード大学教授のハワード・ガードナー(Howard Gardner)は、この論文について「たぶん今世紀で最も重要で、かつ最も有名な修士論文」と評した。

シャノンの情報理論

クロード・エルウッド・シャノン (Claude Elwood Shannon, 1916年4月30日 - 2001年2月24日) はアメリカの電気工学者、数学者。20世紀科学史における、最も影響を与えた科学者の一人である。

- * 情報量を表す単位「ビット/シャノン」とは？
- * シャノンの情報理論 (Information theory)
- * “データの定量化”に関する数学。可能な限り多くのデータを媒体に格納したり通信路で送ったりすることを目的としている。
- * “情報エントロピー” = “データの尺度”
データの格納や通信に必要とされる平均ビット数で表現。
 - * 例えば、日々の天気が3ビットのエントロピーで表されるなら、十分な日数の観測を経て、日々の天気を表現するには「平均で」約3ビット/日 (各ビットの値は0か1) と言うことができる。
 - * “標本化定理”:
 - * アナログデータをデジタルデータへと変換する時、どの程度の間隔でサンプリングすればよいかを定量的に表す方法 (1949年)
 - * その他、暗号理論、チェスを解くプログラムなどの成果も (wikipedia読もう)

1.3 情報量

- * 情報量(じょうほうりょう、**エントロピー**)は、情報理論の概念で、あるできごと(事象)が起きた際、それが**どれほど起こりにくいか**を表す尺度である。
- * 頻繁に起こるできごと(たとえば「犬が人を噛む」)が起こったことを知ってもそれはたいした「情報」にはならないが、逆に滅多に起こらないできごと(たとえば「人が犬を噛む」)が起これば、それはより多くの「情報」を含んでいると考えられる。情報量は**そのできごとがどれだけの情報をもっているかの尺度**であるともみなすことができる。
- * なおここでいう「情報」とは、あくまでそのできごとの起こりにくさ(確率)だけによって決まる**純粹に数学的な量**のことであり、それが**個人・社会にとってどれだけ意義のあるものか**とは無関係である。
- * たとえば「**自分が宝くじに当たった**」事象と「**見知らぬAさんが宝くじに当たった**」事象は、前者の方が**有意義な情報**に見えるが、両者の情報量は全く同じである(宝くじが当たる確率は所与条件一定のもとでは誰でも同じであるから)。

1.3 情報量

- * 選択情報量(自己エントロピー)と平均情報量(エントロピー)
 - * それぞれのできごとの情報量だけでなく、それらのできごとの情報量の平均値を情報量と呼ぶ。
 - * 両者を区別する場合には、前者を選択情報量(自己エントロピーとも)、後者を平均情報量(単にエントロピーとも)と呼ぶ。
- * 事象 E が起こる確率を $P(E)$ とするとき、事象 E が起こったことを知らされたとき受け取る(選択)情報量を

$$\underline{I(E)} = \log \frac{1}{P(E)} = -\log P(E)$$

と定義する。

- ★要は...「**起こりにくい事象**(=生起確率が低い事象)の情報量ほど、**値が大きい**」、そのための **$-\log$ 表現**だと思えば良い。

1.3 情報量

$$I(E) = \log \frac{1}{P(E)} = -\log P(E)$$

- * Logの底が2の場合、 $1/2^n$ の確率で起こる事象の情報量 n はである。
- * 【情報量の加法性】
 - * 情報量には加法性がある。AとBが独立な事象のとき、「AもBも起こる」という事象の情報量は、Aの情報量とBの情報量の和である。
 - * 例えば、52枚のトランプから無作為に1枚を取り出すという試行を考える。「取り出したカードはハートの4である」という事象の情報量は、前述の定義から $\log 52$ であると分かる(カードが52枚なので、ハートの4も $1/52$ ですね)。
 - * [確認] ここで、「取り出したカードのスイートはハートである」という事象と「取り出したカードの数字は4である」という事象の2つを考えると、
 - * 前者の情報量は $\log 4$ (カードのスイートは4種類)
 - * 後者は $\log 13$ である (カードの数字は1~13の13種類)
 - * この両者の和は $\log 4 + \log 13 = \log(4 \times 13) = \log 52$ となり、「取り出したカードはハートの4である」という事象の情報量と等しい。

1.4 符号化

- * 2つの「Shannonの定理」
- * 「シャノンの第一基本定理」
 - * ノイズ(雑音)がない通信路で効率よく情報を伝送するための符号化「情報源符号化定理」ともいう
 - * よく出てくる情報源記号には短い符号、あまり出てこない情報には逆を。
- * 「シャノンの第二基本定理」
 - * ノイズがある通信路で正確に情報を伝送するための誤り訂正符号。「通信路符号化定理」ともいう。
単一通信路あたりの伝送容量に上限があることを意味する。
- * これらはそれぞれデータ圧縮の分野と誤り訂正符号の分野の基礎理論となっている。

1.4 符号化

* ハフマン符号 (Huffman coding)

- * 1952年にデビット・ハフマンによって開発された符号。
- * コンパクト符号やエントロピー符号の一つ。
- * JPEGやZIP (Deflate) などの圧縮フォーマットで使用されている。
- * シヤノン符号化が最適ではない場合が存在する不完全な符号であったのに対し、ハフマン符号は(整数の符号語長という制約のもとでは、)常に最適な符号を構成できる。擬似的に実数の符号語長を割り振る算術符号と比較すれば、データ圧縮効率は劣る。ただし、算術符号やその他の高効率の符号化法と異なり、特許の問題が無い。

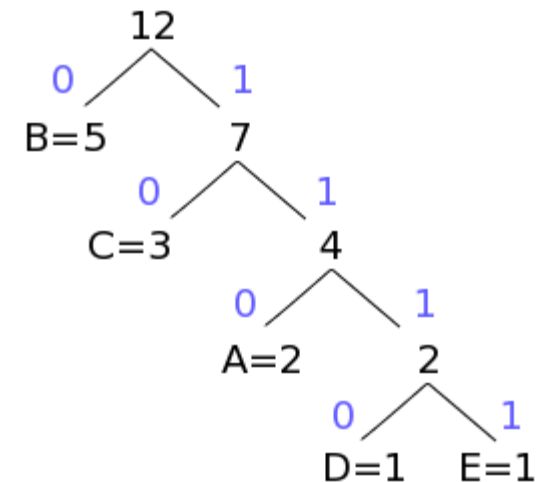
1.4 符号化

* ハフマン符号 (Huffman coding)

- * 1952年にデビット・ハフマン (英語版) によって開発された符号である。コンパクト符号やエントロピー符号の一つ。JPEGやZIP (Deflate) などの圧縮フォーマットで使用されている。
- * **データに出現する記号の個数**を求める。それが木構造の葉に相当すると見なし、ボトムアップで木を構成する。

* 入力「DAEBCBACBBBC」に対する例

| 文字 | 個数 | 符号 |
|----|----|------|
| B | 5 | 0 |
| C | 3 | 10 |
| A | 2 | 110 |
| D | 1 | 1110 |
| E | 1 | 1111 |



出現頻度と割り当てられた符号

第3回予告

- * ここまでは教科書の7ページの内容です
- * シャノンの圧縮もやってみてください！