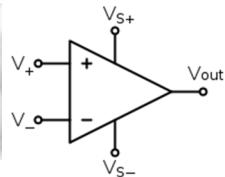


前回の復習: アナログ信号処理: 演算増幅器(オペアンプ)

- ・ アナログ信号処理のコアデバイスの一つ。
- ・ 入力信号の n 倍、加算、微分、積分といった処理が係数のコントロールも含めて正確にできる。
- ・ V_+ と V_- の差分が増幅されて出力される。

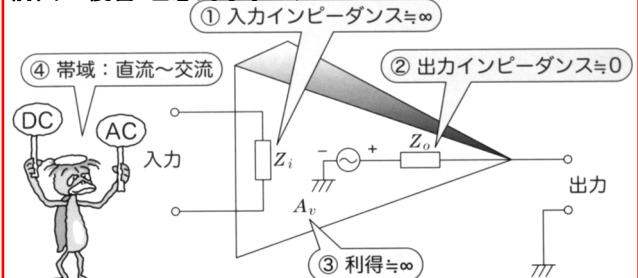
オペアンプの用途:

- ・ 加算・減算回路
- ・ 微分・積分回路
- ・ 発振回路
- ・ フィルター回路
- ・ コンパレータ回路
- ・ センサ回路
- ・ 制御回路
- ・ 変換回路 (電流-電圧変換)



参考書: オペアンプの基礎マスター
(電気書院) ISBN:9784485610015

前回の復習: 理想的なオペアンプ:



現実のオペアンプ:

- ① 入力インピーダンスは $\infty \Omega$ ではなく、 $10^6 \sim 10^{14} \Omega$ 程度。
- ② 出力インピーダンスは 0Ω ではなく、 100Ω 以下。
- ③ 利得は無限大ではなく、 $10^5 \sim 10^7$ 程度。
- ④ 周波数帯域は無制限ではなく、直流～数十MHz程度。

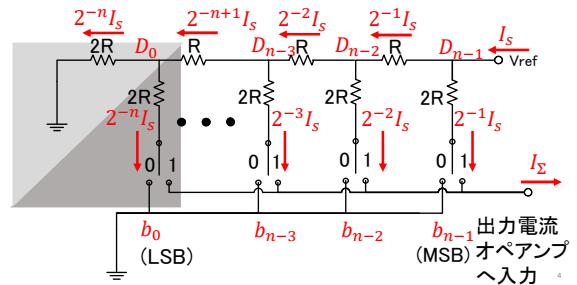
前回の復習: ディジタル信号処理: 現代の計測の主流

- ・ 利点
 - ・ 表現精度がビット数でスケーラブル。
 - ・ 同じ特性を実現可能。
 - ・ Central Processing Unit: CPU, Digital Signal Processor: DSP, Field-Programmable Gate Array: FPGAなどの電子デバイスにより多彩な処理が可能。
 - ・ 集積化・システム化が容易。
 - ・ ディジタル通信・ストレージとの相性が良い。
- ・ 欠点(注意点)
 - ・ 量子化誤差に注意。
 - ・ サンプリング周波数に注意。
 - ・ 過大負荷になると処理速度が間に合わない可能性。

前回の復習: はしご形R-2R方式:

n 種類の倍々の抵抗を用意する代わりに2種類の抵抗(Rと2R)だけで、D/A変換器2の左部分と同等の回路を構成する。精度が良い。

$$V_o = -V_r \left(\frac{b_{n-1}}{2^1} + \frac{b_{n-2}}{2^2} + \cdots + \frac{b_1}{2^{n-1}} + \frac{b_0}{2^n} \right) = -V_r D,$$



アナログ/デジタル(A/D)変換器の種類:

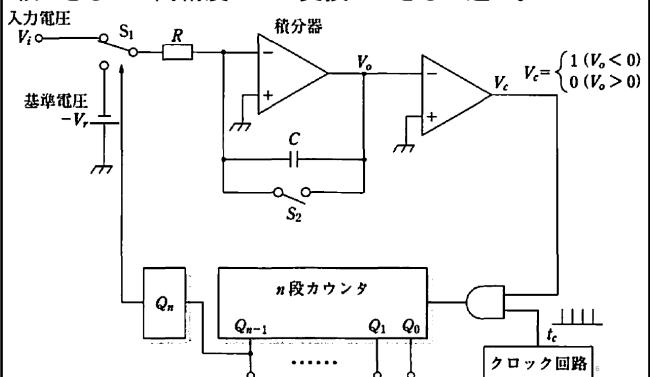
- ・ 積分型
 - 二重積分型: 変換速度は遅いが高精度である。
⇒デジタル電圧計
- ・ 比較型
 - 逐次比較型(D/A変換活用): 変換速度は普通で最も普及している。
⇒汎用
 - 並列比較型: 変換速度は速いが、ビット数が増えると回路が複雑になる。
⇒ビデオ信号処理

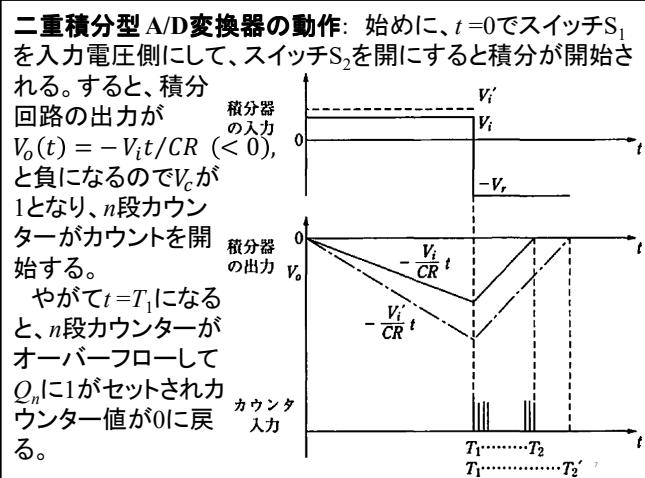
A/D変換器の特性:

- ・ 分解能: (フルスケール)/(離散化の単位)。
16ビットや24ビットのものがある。
- ・ サンプリング周波数: 1秒間に変換を行う回数。
時間方向の分解能。
- ・ 誤差: 電圧方向と時間方向

二重積分型 A/D変換器:

入力電圧 V_i と基準電圧 $-V_r$ を積分器でそれぞれ積分して比較する。積分回路の特性を相殺できるので高精度のA/D変換ができるが遅い。





この時点での経過時間は $t=T_1=2^n \cdot t_c$ で、積分回路の出力 $V_o(t)$ は、

$$V_o(T_1) = -\frac{V_i}{CR} T_1 = -\frac{V_i}{CR} 2^n t_c,$$

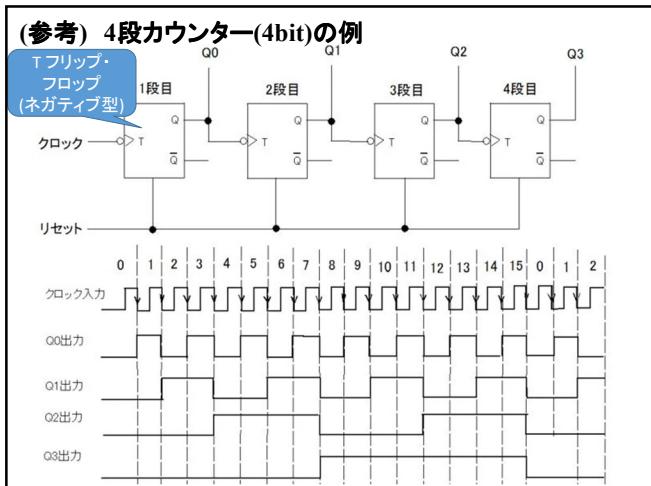
となる。

Q_n が 1 になると同時にスイッチ S_1 が(最大)基準電圧側 ($-V_r$) に切り替わり、今度は基準電圧の積分が開始されて、再び 0 からカウントが始まる。基準電圧は負なので、 V_o は正の方向へ動き、やがて $t=T_2$ で 0 になる。すると、 V_c が 0 となりカウントが止まる。その時点でのカウント数を N とすると、 $T_2-T_1=N \cdot t_c$ なので、

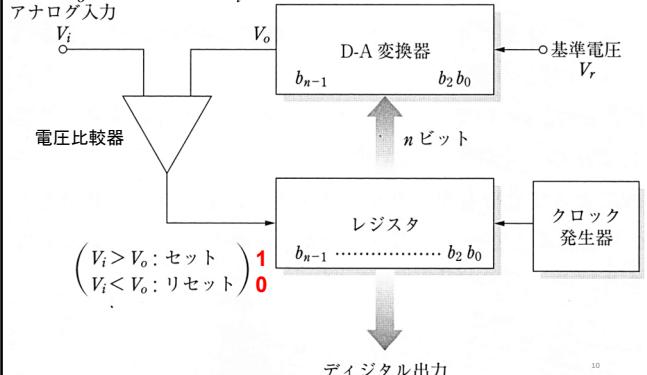
$$\frac{V_i T_1}{CR} = \frac{V_r}{CR} (T_2 - T_1), \quad V_i 2^n t_c = V_r N t_c, \quad V_i = \frac{N}{2^n} V_r,$$

となる。従って、 N がそのまま小数 D の n ビットデジタル量に相当する。

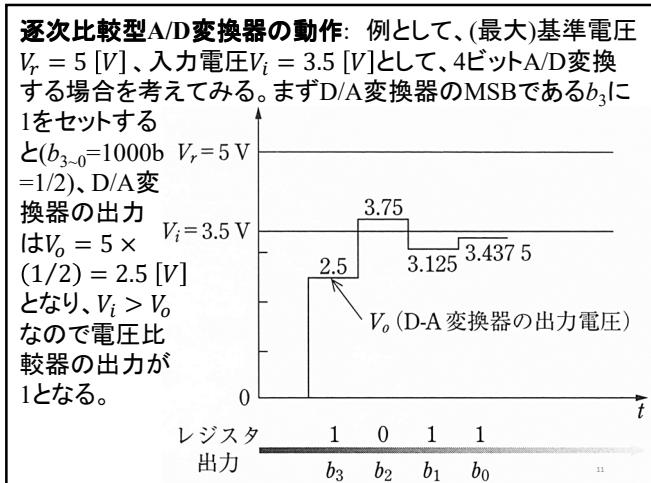
8



逐次比較型A/D変換器: 前回学習したD/A変換器の各ビットに最上位ビット(MSB)から順に1をセットしてその出力電圧 V_o と、入力電圧 V_i と比較してデジタル値を決めていく。アナログ入力



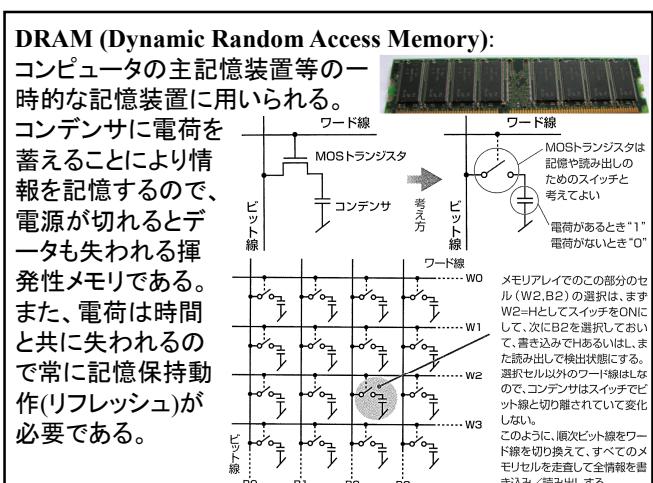
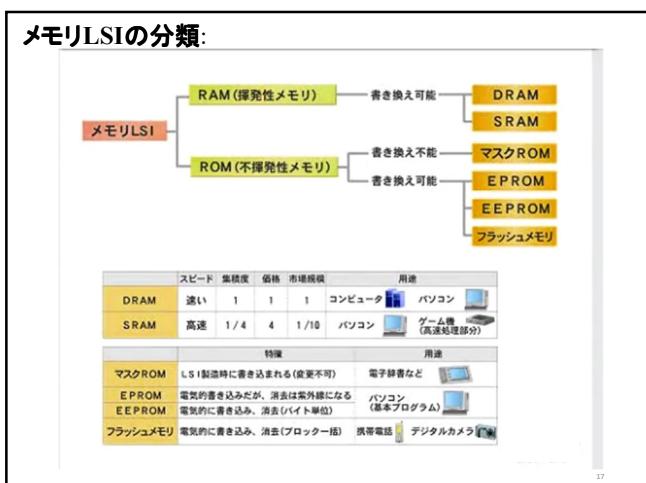
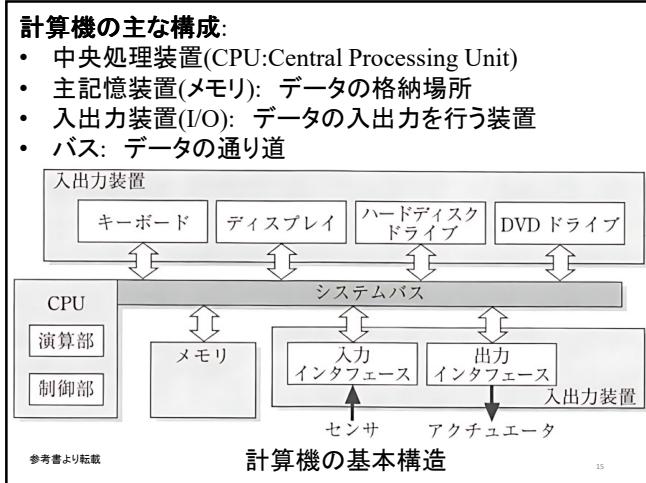
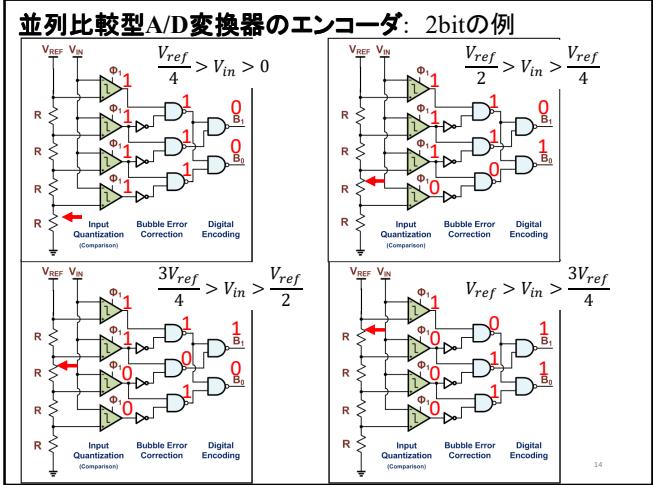
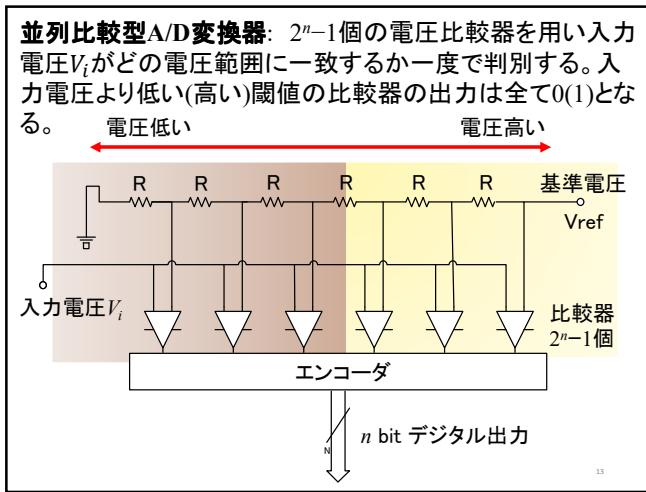
10



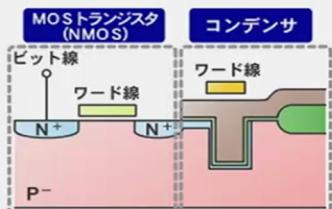
従って、 $b_3=1$ がセットされる。次に b_2 に 1 をセットすると ($b_{3-0}=1100_b=3/4$)、 $V_o = 5 \times (3/4) = 3.75 [V]$ となり、 $V_i < V_o$ なので電圧比較器の出力が 0 となり、 $b_2=0$ がセットされる。次に b_1 に 1 をセットすると ($b_{3-0}=1010_b=5/8$)、 $V_o = 5 \times (5/8) = 3.125 [V]$ となり、 $V_i > V_o$ なので電圧比較器の出力が 1 となり、 $b_1=1$ がセットされる。最後に b_0 に 1 をセットすると ($b_{3-0}=1011_b=11/16$)、 $V_o = 5 \times (11/16) = 3.4375 [V]$ となり、 $V_i > V_o$ なので電圧比較器の出力が 1 となり、 $b_0=1$ がセットされる。最終的に $b_{3-0}=1011_b$ がA/D変換値となる。

$$5 \times \left(\frac{1}{2} + \frac{0}{4} + \frac{1}{8} + \frac{1}{16} \right) = 5 \times \frac{11}{16} = 3.4374 [V] \cong 3.5 [V]$$

12



DRAMのしくみ:



メモリセル

DRAMが1ビットの情報を記憶する単位構造

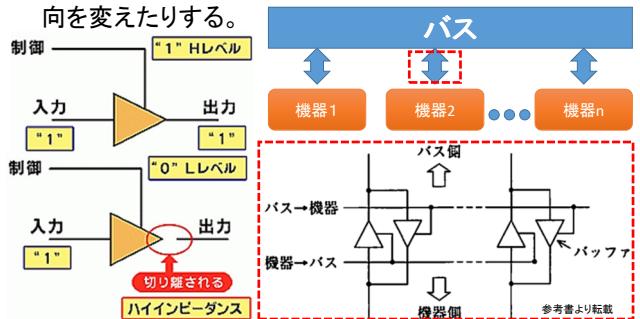
システムバスとアドレス:

- CPUはバスを通して、メモリや入出力装置とデータをやり取りする。その際には、目的のデータが格納されている場所をアドレスで指定する必要がある。
- メモリ: データが格納されている場所をメモリアドレスで指定する。メモリアドレス全体のことをメモリ空間ともいう。
- 入出力機器: I/Oポートと呼ばれるアドレスで指定する。
 - メモリアドレスから独立した入出力機器接続専用の信号線を用いてアドレス指定(I/O mapped I/O)
 - メモリアドレスの一部を入出力機器のI/Oポートに割り当てるメモリアクセスと同様の方法でアドレス指定(memory mapped I/O)⇒現在は、これが主流。

20

バスを通じたデータのやりとり:

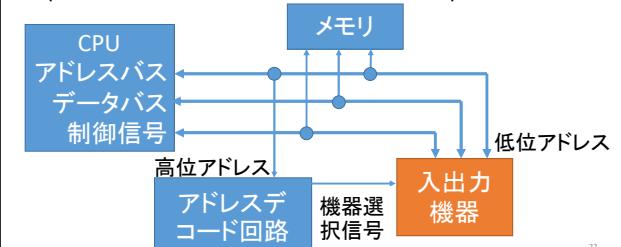
- システムバスのうち、アドレス指定用の信号線をアドレスバス、データ転送用の信号線をデータバスという。
- 一つのバスに多くの機器を接続させるため、3ステートバッファを用いて、バス↔機器間を切り離したり、信号方向を変えたりする。



データの入出力: Memory mapped I/Oの例

CPUとメモリ、入出力機器との接続の模式図。

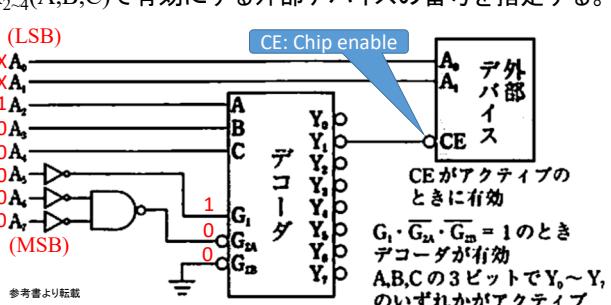
- この例では、アドレスの高位部分が入出力機器を指定している。
- アドレスデコード回路が高位アドレスを解析して、もし入出力機器へのアクセスならば、当該機器へと知らせる。(メモリへのアクセスならば、何もしない)



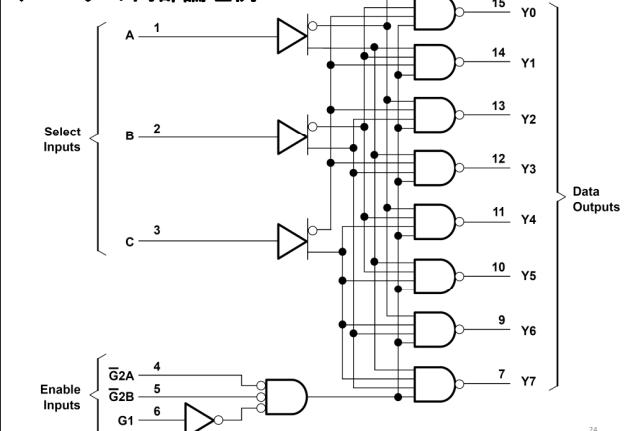
21

アドレスデコード回路の構成例: 74ACH138

4つ(2bit)のアドレスを持つ外部デバイスを下図の様にデコードのY_iに接続すると、CPUの8bitアドレス空間のA_{0~7}=04h~07hで、この外部デバイスにアクセスできる。A_{2~4}(A,B,C)で有効にする外部デバイスの番号を指定する。



デコーダの内部論理例: 74ACH138



22

デコーダの真理値表の例: 74ACH138

OUTPUTSが‘L’になっているところが、選択された入出力機器。

FUNCTION TABLE													
ENABLE INPUTS		SELECT INPUTS		OUTPUTS									
G1	$\overline{G2A}$	$\overline{G2B}$	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	X	H	H	H	H	H	H	H	H
X	X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	L	H	L	H	H	H	L	H	H	H	H
H	L	L	H	L	L	H	H	H	H	L	H	H	H
H	L	L	H	H	L	H	H	H	H	H	L	H	H
H	L	L	H	H	H	H	H	H	H	H	H	H	L

25

- アドレスデコーダは \overline{AS} がLになったら、 A_n の高位アドレスをデコードして該当する外部デバイスのCEをHにして、選択されていることを通知する。
- 選択された外部デバイスは R/W がHであることを確かめ、 D_n にデータを送出して、 \overline{DTACK} をL(アクティブ)にする。
- CPUは \overline{DTACK} がLになったら、 D_n からデータを取り込み始め、取り込みが完了したら \overline{AS} をH(非アクティブ)にする。
- \overline{AS} がHになったら、アドレスデコーダが外部デバイスのCEをLにするので、外部デバイスがデータ送信を止めて \overline{DTACK} をH(非アクティブ)にする。

このように \overline{DTACK} 信号で互いの状態を確認しながら、データの送受信を行う方式を非同期通信という。

27

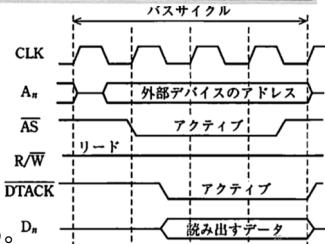
制御信号の例: 文字の上に線があるのはLOWでアクティブ

信号	内容
\overline{AS}	アドレスストローブ。アドレスバスに有効なアドレスが出力されていることを示す。
R/W	リード・ライト。データ転送がリード(読出し)なのかライト(書き込み)なのかを定める。highのときリード、lowのときライトである。
DTACK	データ転送アケノレッジ。外部デバイスがデータをリード、またはライトできる状態をCPUに知らせるための信号である。アクセスタイムの遅い外部デバイスとのデータのやりとりに用いられる。
INT	割込み要求。外部デバイスから割込み要求を伝えるために用いられる。

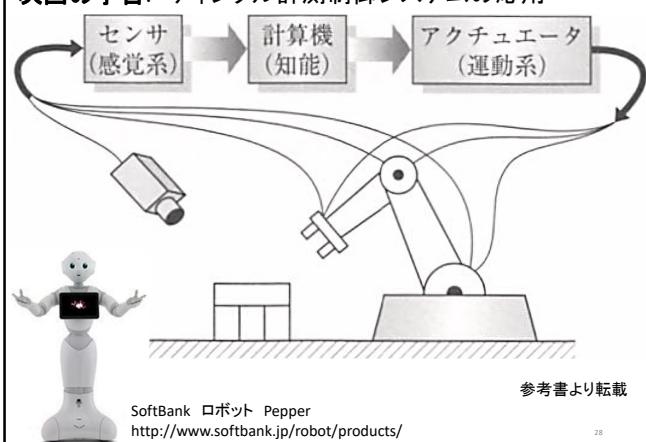
データ読み込みタイミング

チャートの例: 外部デバイスからデータを読み込む場合。

- CPUが R/W をH(リード)にして、 A_n に外部デバイスのアドレスを送出して、 \overline{AS} をL(アクティブ)にする。その後 \overline{DTACK} を監視している。 D_n にデータを読み出すデータ



次回の予告: デジタル計測制御システムの応用



28

(参考)デルタ・シグマ形A/D変換器: 入力信号を1bit(1または-1)にデジタル変換した時の量子化誤差を積分器で蓄積しながらフィードバックすると、出力値の数が入力信号の大きさに比例する。つまり、入力信号が+なら1が多く出力され、-なら-1が多く出力される。それがあるサンプリング周波数で足し合わせれば入力信号をデジタル値に変換できる。高精度だが遅い。

