

マイクロ体内ロボットの中核となる組み込みシステムの開発

西村 功[†] 小林 正嗣[†] 吉田 孟浩[†] 武石 学[†]

泉 知論^{†,††} 加藤 俊之[‡] 山内 寛紀^{†,††}

[†]立命館大学大学院 理工学研究科 〒525-8577 滋賀県草津市野路東 1-1-1

^{††}立命館大学 理工学部 電子情報デザイン学科 〒525-8577 滋賀県草津市野路東 1-1-1

[‡]日本ケイデンス・デザイン・システムズ社 〒222-0033 神奈川県横浜市港北区新横浜 3-17-6

E-mail: [†] {ro010014, ro006014, re013020, re007039}@se.ritsumeai.ac.jp

^{††} {t-izumi, yamauchi}@se.ritsumeai.ac.jp

あらまし 体腔内を自由に移動し、情報収集・診断・治療を行う医療用マイクロ体内ロボットの中核となる組み込みコンピュータシステム(体内コンピュータ)の開発を行っている。この体内コンピュータの実現には極低消費電力化と高度の安全性が鍵である。本研究では、体内ロボット専用組み込みシステムの仕様を検討し、ハードウェア及びソフトウェアの設計・検証、開発環境の構築を行う。種々の診断・治療、さらには家畜用ロボットなど、様々な目的のマイクロロボットの共通プラットフォームとなる汎用性・拡張性の高いシステムを目指している。

キーワード マイクロ体内ロボット, 体内コンピュータ, 組み込みシステム, マイクロプロセッサ, 組み込み OS

Design of Embedded System for Micro-Capsule-Robot

Tsutomu Nishimura[†] Masatsugu Kobayashi[†] Takehiro Yoshida[†] Manabu Takeishi[†]

Tomonori Izumi[†] Toshiyuki Kato[†] and Hironori Yamauchi[†]

[†]Graduate School of Science and Engineering, Ritsumeikan University

1-1-1, Noji-higashi, Kusatsu, Shiga, 525-8577 Japan

^{††}Department of VLSI System Design, College of Science and Engineering, Ritsumeikan University

1-1-1, Noji-higashi, Kusatsu, Shiga, 525-8577 Japan

[‡]Cadence Design Systems, Japan 3-17-6 Sinyokohama, Kouhoku-ku, Yokohama-shi, Kanagawa 222-0033 Japan

E-mail: [†] {ro010014, ro006014, re013020, re007039}@se.ritsumeai.ac.jp

^{††} {t-izumi, yamauchi}@se.ritsumeai.ac.jp

Abstract We develop a computer platform as a central system of the micro-capsule-robot, which moves freely inside a body, and gets information about the body for diagnosis. This computer controls the micro-robot, and processes the packet communication and image compression. Hence extremely small power and high safety are important for realizing. In this paper, we plan a computer architecture suits for the micro-robot, and design its hardware and software. It aims at the high generality and extendibility system. And it becomes a common platform of target a variety of the micro-capsule-robot.

Keyword Micro-capsule-robot, computer platform, embedded system, micro-processor, embedded operating system

1. はじめに

近年、患者に大きな負担をかけることなく診断・治療を行う、いわゆる低侵襲診断・治療の新技术の1つとしてマイクロ体内ロボットが注目されている^[1]。マイクロ体内ロボットとは、コンピュータを組み込んだ医療用カプセル型ロボットを示す。このロボットを飲み込むと、外部サーバからの遠隔操作に従い体内を循環し、情報収集や医療診断、さらには病巣治療まで行えるという革新的な医療の実現を目指している(図1)。

マイクロ体内ロボットは世界中で実現に向けて研究・開発がなされており、今後さらなる発展が期待されている。

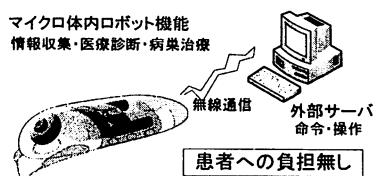


図1 マイクロ体内ロボットイメージ

現在のマイクロ体内ロボットは体内を自然に循環する浮遊型が主流であり、体内画像を撮影する機能のみを持つ浮遊型ロボットは、一部で実現・利用がなされている。将来的に、体外から移動操作可能である滞留型や、治療操作可能である可動型への発展が期待されている。そのためには、映像の高品質化、通信の高精度化や低消費電力化、小型化といった点での技術の向上が必要である^{[2][3][4]}。

本研究ではマイクロ体内ロボットに向けた、高い汎用性および拡張性を持つ共通プラットフォームの実現を目指している。そこで我々は、共通プラットフォームのプロトタイプの様式検討、設計を進めている。

本稿では低消費電力と高安全性に配慮し、今後への幅広い拡張性を持たせた体内コンピュータプラットフォームのプロトタイプを提案した。また HW 及び SW の設計、開発環境の構築、そしてそれらの検証を行った。

第 1 章で概要紹介を行っている。以下第 2 章で体内コンピュータの構成・機能について述べ、第 3 章で搭載するマイクロプロセッサについて、第 4 章で設計および検証について述べる。最後に第 5 章でまとめを述べる。

2. 体内コンピュータ構成・機能

マイクロ体内ロボットは、無線で動作するロボットであるため大きな電力供給ができないこと、体内に入るためにロボットサイズが小さいこと、医療用という特殊な用途であることなどの条件がある。これらのことから体内コンピュータに求められる要件は

- 極低消費電力
- 小回路規模
- 高度な安全性

の 3 つが挙げられる。これらの要件に従って各モジュールの開発を行っている。

2.1. 体内コンピュータ概要

体内コンピュータとは、マイクロ体内ロボットの頭脳部分を示す。体内コンピュータが必要とするのは以下の機能である。

- マイクロ体内ロボット制御機能
- 体外サーバとの無線通信機能
- 動画像撮影機能
- 各種センサ機能 (温度センサ, 圧力センサ, 加速度センサ等)
- 各種ドライブ機能 (移動用 / 治療用駆動モジュール等)

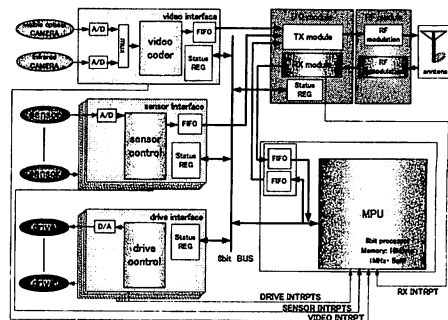


図 2 体内コンピュータ概要図

制御機能と無線通信機能は体内コンピュータの必須機能である。したがって、制御機能を実現するマイクロプロセッサ (MPU) と、無線通信機能を実現する通信モジュール (I/O module) の 2 モジュールが体内コンピュータの基盤構成要素であり、体内コンピュータプラットフォームと呼んでいる。このプラットフォームの上に、目的に応じて動画像撮影機能・各種センサ機能・各種ドライブ機能を実現する拡張モジュールを搭載可能な仕様とする。体内コンピュータの概要図を図 2 に示す。図 2 の中でセンサそのものやドライブ機能の駆動デバイス部分を除いた全ての回路をシングルチップで実現することを目標としている。本節ではこれらモジュールについて概要を説明する。ただし、センサ・ドライブモジュールについては詳細が決まっていない段階なので省略する。

2.1.1. マイクロ体内ロボット制御機能

制御機能は MPU が核となって実現する。MPU は、周辺モジュールであるインタフェースモジュールを制御する。これらインタフェースモジュールはセンサや駆動デバイスを制御を担当している。従って、MPU はインタフェースモジュールを介して、間接的にマイクロ体内ロボット全体を制御することとなる。

各モジュールの制御は、モジュールが持つステータスレジスタ (Status REG) にパラメータを書き込むことで実現する。これらステータスレジスタは MPU のデータメモリと共用のバスで接続されており、メモリマップされている。その為、MPU から見ると、パラメータの読み書きはメモリへの標準アクセスと同じ方法で行える。周辺モジュールにステータスレジスタを用意し、データバスに接続するだけで、MPU はそのモジュールをソフトウェア制御することができるようになる。制御方法を特別に設けず統一することにより、回路規模の削減を行い、さらに高拡張性を有するアーキテクチャを実現している。

体外サーバから送られてくるデータは体内ロボットの挙動をコントロールするための制御命令である。MPU は通信モジュールを介して受け取った制御命令に従って、各モジュールを制御する。

2.1.2. 無線通信機能

体外サーバとの無線通信機能は通信モジュールが実現する。通信データに付加するヘッダハンドルの処理はすべて通信モジュールで行い、MPU は通信モジュールの制御を行う。

さらにカメラやセンサから通信モジュールへと専用線が設けられている。得られた計測データは、MPU を経由せず、直接通信モジュールに送られ、さらにそこから外部サーバへ送られる。

MPU が通信処理そのものを行わないことにより、MPU の機能を周辺モジュールの制御および割込み処理に特定することができ、回路規模の削減につながる。さらに通信処理中に MPU が動作している必要がない。そのため低消費電力化にも貢献する。

本研究ではマイクロ体内ロボット用の通信プロトコルを新たに開発した。マイクロ体内ロボットの通信プロトコルでは、低消費電力だけでなく、医療用という観点からリアルタイム性・送受信データの確実性が求められる。そのため、専用のプロトコルとして計算・通信負荷を抑え、送受信データを確実かつ能率的に移すことを目的としている。このプロトコルに適合したヘッダハンドル処理を行う通信モジュールも別途開発した。それらについての詳細は、本研究会に投稿している論文「マイクロ体内ロボット向け通信モジュールの構成と設計」に記述している。

2.1.3. 動画撮影機能

マイクロ体内ロボットには体内映像を撮影するカメラ、および体内映像の高圧縮処理を低演算量で処理できるビデオモジュール搭載する予定であり、別途開発を行っている^[5]。

撮影・圧縮された映像信号は無線通信により体外サーバへと送られる。この映像通信は、高周波減衰により伝送容量が制限される。しかしその一方、撮影した体内映像が病巣診断やリモート治療に利用できるよう高品質である必要がある。また、長期体内滞留要求とロボットサイズ制約による、徹底した低消費電力も求められる。これらの問題を解決する新映像圧縮アルゴリズムを考案し、ソフトウェアにより実現した。その結果 JPEG・JPEG2000 と比較してほぼ同等の映像品質を、演算量換算 1/3~1/5 で実現した。このことにより消費電力換算でさらなる削減が見込める。現在前述アルゴリズムを基に HW 構築中である。

2.2. モジュール動作モード

極低消費電力を実現するために、各モジュールに動作モードを規定した。動作モードを動的に変更することにより、省電力化を目指す。動作モードには、処理モード、低速処理モード、待機モードの3つを現在想定している。モード遷移により、供給されるクロックの速度が変更される。動作モードは、MPU 内のクロック調整器により実現される。クロック調整器は、体内コンピュータの全てのモジュールのクロックを一元管理するモジュールである。

本システムはイベントドリブン型アーキテクチャを採用しており、体内コンピュータは通常状態を、処理を行わない待機モードとしている。

動作が必要なモジュールは割込み信号を出し、MPU がそれを受け付けると対象モジュールを処理モードへと遷移させる。動作終了時にも割込み信号を出し、待機モードへ遷移させることで、低消費電力を実現する。

MPU 自身の動作モードの遷移は特殊である。内外からの処理命令が無いことを判断すると自動的に、クロック供給を停止させ、待機モードを実現する。MPU は待機モード中、全ての割込みを受付可能である。動作が必要なモジュールは割込み信号を出し、MPU がそれを受け付けて処理モードへと移る。

モジュール毎にモードの選択ができるアーキテクチャにしているので、必要最低限のモジュールのみを動作モードにすることができる。

2.3. 各モジュール性能

体内コンピュータはプラットフォームとしての MPU と通信モジュールを持ち、搭載モジュールとしてビデオモジュールを現在開発中である。さらに拡張モジュールとして、各種センサ機能・ドライブ機能などを計 13 モジュール追加可能である。表 1 に各モジュールの性能表を示す。

表 1 各モジュール性能表

機能	MPU		ビデオモジュール		
	特徴	全体制御		映像圧縮処理	
モード	クロック制御		高圧縮、低演算量		
周波数	処理	待機	処理	低速	待機
	1MHz	—	20MHz	1MHz	—

機能	通信モジュール		拡張モジュール	
	トランシーバ	レシーバ	センサ・ドライブ	
特徴	送信処理	受信処理	温度/圧力・動作制御	
モード	エラー回避		複数拡張接続可能	
周波数	処理	待機	処理	待機
	10MHz	—	10MHz	100kHz以下

計13モジュールまで拡張可能

3. 体内コンピュータ用プロセッサ

3.1. MPU ハードウェアアーキテクチャ

組込みシステムには、文献^[6]に代表されるような様々なプロセッサが開発利用されている。しかし、マイクロ体内ロボットへの搭載には回路規模の低減と極低消費電力化が必須である。そこで本研究では、体内コンピュータ専用のマイクロプロセッサ(MPU)を開発した。

本 MPU は大きく分けて、

1. MPU-Core (専用プロセッサコア)
2. Instruction Memory (命令メモリ) 1word = 16bit
3. Data Memory (データメモリ) 1word = 8 bit
4. Interrupt Controller (割込み処理コントローラ)
5. Timer Unit (タイマーユニット)
6. Clock Controller (クロック調整器)

の 6 部分からなり、さらに MPU-通信モジュール間のデータ制御を行うコントローラ(MPU-IO-Controller)を持つ。本 MPU のブロック図を図 3 に、MPU の諸元を表 2 に示す。

本 MPU への要件は低消費電力化、小回路規模の 2 点である。これらの見地から、MPU アーキテクチャの特徴を以下で述べる。

本 MPU の目標動作周波数は 1MHz であり、無理な高速化の必要はない。そのためパイプライン処理は行わず、全ての命令を 2 サイクルで実行するアーキテクチャを実現した。

本 MPU が扱うデータは、制御データが主である。制御データはデータ量が少なくかつ転送頻度も少ない、という特徴を活かして、命令長を 16bit、データ長を 8bit とした。データのワード幅を、扱うデータのサイズに合わせたことにより、無駄なデータバス使用やデータメモリ使用での電力消費を抑えている。

命令・データのワード幅がそれぞれ異なる為、命令メモリとデータメモリを分離するハーバードアーキテクチャを採用した。命令フェッチが 1 サイクルで実行できることにより、全命令が 2 クロックサイクルで実行できるアーキテクチャを実現した。

また、周辺モジュールの制御が主な仕事であることから、算術演算は多用しないと考えられる。その為、使用頻度の少ない乗算命令およびハードウェアは搭載しないことにした。

本 MPU に必要な機能には以下の 3 つが挙げられる。

- 周辺モジュール制御機能
- 動作モード変更機能
- イベント検知機能

以下でそれぞれの実現方法について説明する。

周辺モジュール制御機能については、メモリマップされたレジスタにパラメータを書き込むことにより実現される。そのため、通常のメモリアクセス命令を使用すれば、周辺モジュールの制御が可能である。制御専用のハードウェアを持たないことにより、回路規模の増大を防ぐことになる。

動作モード変更機能には、体内コンピュータの全モジュールのクロックを管理するモジュール、クロック調整器を搭載することで実現した。クロック調整器は MPU に供給するクロックだけではなく、周波数の異なるビデオやセンサ、ドライブ、通信モジュールといった全てのモジュールのクロックを管理する回路である。動的なクロック管理を行うことにより、不要な動作を行わず、更なる低消費電力動作を実現する。さらに、チップ化した際には各モジュールの電圧制御を行う拡張が可能のように、クロック調整器内にはパワーマネジメント用レジスタを設けている。

イベント検知機能は割込み処理コントローラにより実現している。割込み信号が入ると、割込み処理コントローラ内部の割込みフラグレジスタの、割込み信号に対応するビットをセットする。割込みフラグレジスタはプロセッサコアから読み出し可能である。プロセッサコアが割込み処理やクリティカルセクションの処理を行っている間は、割込みにマスクをかけることができる。さらにソフトウェア割込みも可能である。

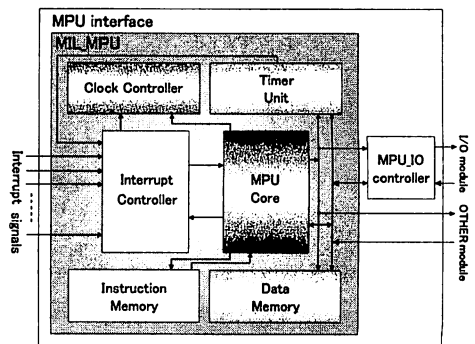
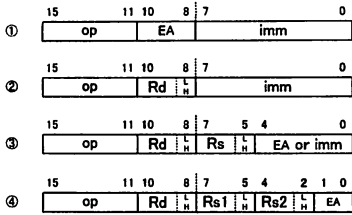


図 3 体内コンピュータ用 MPU のブロック図

表 2 MPU 諸元

アーキテクチャ	8 ビット CISC アーキテクチャ ハーバードアーキテクチャ
命令メモリ	64K words
データメモリ	最大 64K words (1word = 16bit)
レジスタメモリ	最大 64K words (1word = 8bit)
命令数	49 命令
目標動作周波数	1MHz (動作モード時)



op: オペレーションフィールド [5bit], imm: 即値 [8/5bit],
Rd/s L/H: オペランドレジスタ [3bit], EA: 拡張部 [5/3/2bit]

図 4 命令基本フォーマット

3.2. 命令セットアーキテクチャ

小回路規模化・低消費電力化を実現するために、マイクロ体内ロボットの制御に必要なものに限定した命令セットを新たに開発した。

命令は 49 種あり、体内コンピュータの制御を全て網羅できる。データ転送命令、演算命令、分岐命令等の基本的な命令に加え、クロック調整や割り込み制御といった本 MPU 特有の動作は、専用命令として用意している。RISC と CISC を組み合わせたような命令セットとなっている。

命令形式は 1 命令 16bit の固定長であり、命令の基本フォーマットを 4 種類に限定した。基本フォーマットに合わせて命令セットを設計することにより、デコード回路の単純化を行いハードウェア規模の削減を実現している。図 4 に基本フォーマットを示す。

データメモリは 1 ワード 8bit であり、それに対してアドレス幅は 16bit である。汎用レジスタは 16bit のレジスタ 4 本であるが、レジスタの上位 8bit, 下位 8bit のみを独自にアドレス指定可能とし、8bit のレジスタ 8 本としても使用できる。このことにより、アドレッシング方式の種類を増やすことができた。アドレッシング種類として、即値・ベース相対・レジスタ・PC 相対の 4 つを用意している。

3.3. 組込み OS

組込み OS は、処理量を削減するため、極小のシステム構築を行った。周辺モジュール毎にパラメータ用レジスタを持つため、タスク間での同期や通信はほぼ行われない。したがって同期・通信機能を OS では実装していない。本 OS は大きく分けて以下のモジュールから構成される。

- タスク処理
- 割り込み処理
- メモリ管理
- タイマー管理

OS で実現しているシステムコールの一覧を表 3 に示す。

タスクは実行時間が極めて短い処理単位であり、処理が開始されると終了まで、割り込み時以外は中断されることなく実行される。そのためタスクスケジューリングの必要は無いものとし、登録された順番でタスク処理を行う FIFO 型のタスクキューを採用した。このことによりタスク処理のオーバーヘッドを減らしている。

メモリ管理はデータメモリのヒープ領域を動的に管理するプログラムであり、確保と解放から構成される。メモリ管理アルゴリズムはファーストフィットを採用している。メモリの動的な確保要求が頻繁には起こらないという想定から、単純で高速な本アルゴリズムを採用した。

タイマー管理は MPU に用意されているタイマーモジュールを利用したモジュールである。ミリ秒を単位として、65536 ミリ秒までの設定時間後に、ソフトウェアモジュールを実行させるという設定が可能である。

組込みシステム全体のフローチャートを図 5 に示す。本システムは割り込みにより起動されるイベントドリブンのアーキテクチャを採用している。割り込み処理中は割り込みマスクをかけ、多重割り込みは受け付けない。緊急性のない処理に関してはタスクとして処理を行うことにより、割り込み処理時間・割り込み禁止時間を短くする。そのため ISR ではタスクの登録および緊急の処理のみを行うこととする。

表 3 システムコール一覧

名称	説明
task_gen	タスクを生成しタスクキューへ登録する
task_exit	タスクの処理を終了し、タスクを削除する
isr_set	ISRの登録
imctrl_set	割り込みマスクをかける
imctrl_clr	割り込みマスクをはずす
memman_alloc	メモリ領域の確保
memman_free	メモリ領域の解放
timer_set	タイマーの設定

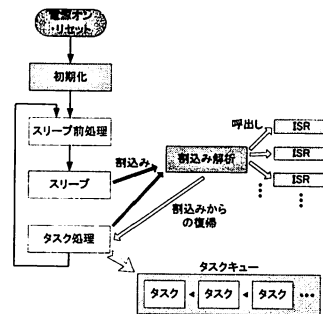


図 5 組込みソフトウェアフローチャート

4. 設計と検証・評価

本稿で提案している体内コンピュータ用プロセッサおよび組込み OS のプロトタイプを設計した。それぞれの設計・評価の結果を表 4 に示す。

ハードウェアの設計には Verilog-HDL を用いた。そしてその設計データを 3 つの環境で動作させ、その結果を比較することにより動作検証を行った。3 つの環境による動作とは、RTL でのシミュレーション、FPGA 上での動作、ソフトウェアエミュレータでのエミュレーションである。RTL でのシミュレーションには Cadence 社の NC-verilog (バージョン 5.1) を用いた。ソフトウェアエミュレータは C++ 言語で設計された命令サイクルで動作するエミュレータであり、PC 上でエミュレーションが可能である。

一方、組込み OS のプロトタイプをアセンブリ言語で設計し、ソフトウェアエミュレータを用いて動作検証を行った。ソフトウェア開発環境として、マクロアセンブラを開発した。開発効率を上げるため、定数の使用や、if 文、for 文、while 文など制御構文も提供している。

4.1. MPU ハードウェア検証・評価

MPU プロトタイプの動作検証を上記の環境で行った。検証パターンとして、体内コンピュータの制御を想定した命令列を作成した。さらにランダムな命令列を生成し、想定していない命令に対してどのように動作するのかを検証した。その結果、各環境で正しく動作することを確認した。

また、FPGA 実装のための論理合成時に算出された動作周波数の最大値は約 40MHz であり、目標としている動作周波数 1MHz を達成した。そして FPGA 上で 1MHz のクロックで正しく動作することを確認した。

さらに、プロセッサコアを論理合成した結果、2 入力 NAND 換算で約 2,500 ゲートという回路規模を得ることができた。

4.2. ソフトウェア検証・評価

組込み OS プロトタイプの動作検証をソフトウェアエミュレータ上で行った。周辺モジュール用ドライバソフトウェアを想定したサンプルプログラムを作成し、正常に動作していることを確認した。

設計した OS プロトタイプの規模は命令数で 1318 ワードである。アドレス幅は 16 ビットであるので、搭載可能なメモリの最大値は 64K ワードである。拡張モジュールのドライバソフトウェアを各 2K ワードと大きく見積もって計算しても、ソフトウェア規模は合計約 30K ワードである。このことから十分実現可能な命令セットアーキテクチャであることを確認した。

表 4 設計・評価結果

ハードウェア評価結果	
動作周波数	マイクロプロセッサ
動作周波数	約40MHz (FPGA用論理合成時)
ゲート数	2,500ゲート(2入力NAND換算)

ソフトウェア評価結果	
命令数	組込みOSプロトタイプ
命令数	1,318ワード

5. まとめ

本稿では、様々な目的を持つマイクロ体内ロボットに標準搭載可能となる体内コンピュータの実現を志向し、幅広い拡張性と高度の安全性を持ち合わせた体内コンピュータプラットフォームのプロトタイプを提案した。

HDL によるプロセッサ開発およびその検証により、プロセッサコアで約 2,500 ゲートという小規模回路を実現し、目標周波数である 1MHz で動作することを確認した。さらに本プロセッサ上で動作する組込み OS を開発し、その動作検証を行った。これらのことにより周辺モジュールを搭載するためのプラットフォームを実現することができた。

予定として、今年度中に周辺モジュールの HW・SW 開発を行う。その後、体内コンピュータ全体のシステムを FPGA 上で実現し、その動作検証を行う。また、電力見積もりを行い、さらなる回路規模の低減と低消費電力化を目指す。

文 献

- [1] <http://www.shigaplaza.or.jp/area/>
- [2] Takakuni Douseki, NTT Microsystem Integration Laboratories, NTT Corporation :Ultralow-voltage MTCMOS/SOI Technology for Batteryless Mobile System
- [3] Takakuni Douseki, NTT Microsystem Integration Laboratories, NTT Corporation :A BATTERYLESS OPTICAL-WIRELESS SYSTEM WITH WHITE-LED ILLUMINATION
- [4] Y. Kado, Y. Matsuya, S. Mutoh, J. Terada, H. Morisawa, Y. Sato, T. Douseki and H. Kyurag, NTT Microsystem Integration Laboratories, : Ultralow-power CMOS/SOI Circuit Technology for Ubiquitous Communications, Extended Abstracts of the 2003 International Conference on Solid State Devices and Materials, Tokyo, 2003, pp746-747
- [5] 坂井田 稔, 豊留 直樹, 泉 知論, 山内 寛紀, 来見 良誠: マイクロ体内ロボット搭載に適した極低演算ビデオ圧縮アルゴリズムの提案, 映像情報メディア学会誌, 2006 Vol60 No.7 pp1114(136)-1118(140)
- [6] Patterson &Hennessy: コンピュータの構成と設計 第 2 版上下, 日経 BP 社, 1999