# 計算機システムⅡ　マイコンでのシステムプログラミング入門

## 1. ArduinoマイコンボードとArduino開発環境による体感情報処理入門

Arduinoは8ビットマイコンを電源やインターフェースの回路とともに小さなボードに搭載したもので，ピンつきワイヤや拡張基盤で簡単に入出力回路を接続でき，適度にハードウェアを抽象化したライブラリを利用してC/C++言語による組み込みプログラミングを意識せずに行うことができる．電子工学，コンピュータに詳しくなくても，さまざまな自動機械，情報機器，アート作品などの製作に活用できるため，世界中で幅広く使われている．情報工学の基礎として，コンピュータの動作原理と入出力のしくみを学ぶのに非常に適しているといえる．このArduinoとArduino開発環境の利用のしかたを，ごく初歩的な例を使って学ぶ．

## 2. ArduinoによるGnu C クロスコンパイラプログラミング入門

プログラミングやコンパイルを行うコンピュータ（ホスト）と，コンパイラの生成したバイナリコードを実行するコンピュータ（ターゲット）が異なる開発は，クロス開発とよばれる．組み込みマイコンやビデオゲーム機では，ターゲット機器上でＯＳやコンパイラ，プログラミング環境を使う余裕がないため，汎用のＰＣなどでプログラミングを行い，ＰＣに接続したプログラマあるいはデバッガとよばれる機器でマイコンのメモリにバイナリコードを書き込み，動作させる．複数の機器を用意し，手順を踏んで行う必要があるため，ホストとターゲットが同一のセルフ開発にくらべ，敷居が高いといえる．

Arduinoでは，プログラマの機能をＵＳＢシリアル変換インターフェースとＣＰＵに書き込み済みのブートローダが行っている．またライブラリでハードウェアを抽象化し，利用者が書くのは短いスケッチ（プログラム）だけにして，デバッグの必要性を下げ敷居を下げている．しかしハードウェアの機能を使い切るような開発はやりにくい．デバッガやGNU C以外のコンパイラ，アセンブラを使用することもできない．

## 3. Arduino を使ったAVR assemblerプログラミング入門

機械語あるいはアセンブリ言語でプログラミングをするのは骨が折れることである．しかし，計算機の基本的な動作を理解するには，命令を直接使うアセンブラのほうが向いている．とくに，ポインタや関数の呼び出し，割り込みやリセット動作などがどのように行われているかを理解するには，C言語を使うより向いている．もちろん，プログラミングはＣＰＵの違いに大きく左右されるが，ほとんどのＣＰＵに共通するごく基本的な考え方があるのも事実である．今日でも，最適化，とくにマイクロ秒単位の動作を扱うためには，アセンブリが使われるケースが少なくない．