

写真1 FRAGILE005RT

# FRAGILE005RT 大解剖!

## ソフトウェア編

こうの じゅんや  
河野 純也

### はじめに

ロボットを作っている方々の多くは、機械加工、または電子回路、あるいはソフトウェアなどといった専門分野をお持ちだと思うが、自分の専門分野以外の実力をいかに伸ばすかが、ロボットを作るためには重要であると考えている。

さて、今回は「FRAGILE005RT (以下、FRAGILE)」(写真1)のハードウェア(機械屋のテリトリー)について解説したが、今回はソフトウェア編ということで、私の苦手とする制御の領域の話となる。苦手なりに何を考えて作ったのか、その考え方を主に解説したいと思う。

なお、間違いや勘違いが多々あると思われるが、それらを含めて参考になればと考えているので、ご容赦願いたい。

### 開発環境

FRAGILEは、CPUに「H8/3694」(ルネサス テクノロジ)を使用しているが、ソフトウェアはアセンブラで記述している。現在は組み込み機器のソフトウェアも、C言語などに代表される高級言語を使うのが一般的であるが、私はアセンブラを使っている。これは、ハードウェアを直接制御している感じがして……といった好みの問題であるが、実行速度も、メモリ消費量も、最適化度合いもすべて書き手次第だという点も気に入っている。

さらにもう1つの理由として、アセンブラで開発している人が少ないことが挙げられる。これは「他に使っている人が少ない」→「サンプルが少ない」→「自分で一から作る必要がある」→「スキルが上

がる(可能性が高い)」といった考えからである。サンプルを参考にするのは完成への近道であるが、苦勞することもロボット作りの目的の1つであると考えている。

### 基本構成

FRAGILEの主要な制御は、H8/3694のタイマAを用いた割り込みルーチン内で行っている。メインルーチンは、メモリとI/Oの初期化、センサーしきい値調整を実行したあとは、何もせずに無限ループして、ただタイマA割り込みが掛かるのを待っているだけである(図1)。

タイマA割り込みが発生すると、モード(競技の際には、コース探索モードと高速走行モードが使用される)に応じて、ラ

イン追従制御、コース記録・再生制御、車速制御などの各処理を実行する。実行後は、前述の無限ループへと戻る。

このほかに、パラメータダウンロード、データアップロード、加速性能測定、制動距離測定など、開発時にのみ実行される処理もある。

### ライン/マーカーセンサー

ライントレースロボットのもっとも重要な部分の1つであるライン/マーカーセンサーには、ハードウェア編で説明したように、「GP2S40」(SHARP)を5個使用している。ただし、5個それぞれの出力特性は必ずしも同じではないため、あらかじめ、黒色、灰色、白色の3条件下で

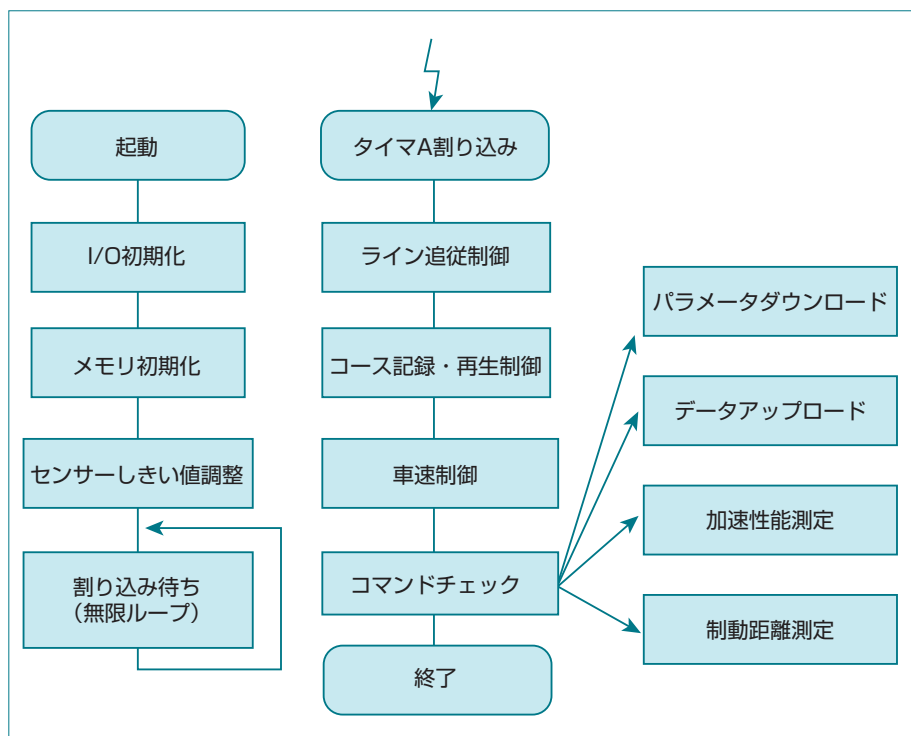


図1 基本構成