

機器制御ソフトウェア用テストツールの開発

Development of Testing Tool for Embedded System Software

渡辺 智* 小寺 哲博* 大塚 勝*
Watanabe, Satoshi Kodera, Tetsuhiro Ootsuka, Masaru

We have developed a tool to test embedded system software, and tentatively it used in control program test of finisher for copy machine. As a result, the following advantages were recognized ; (1) decrease of working process for test (2) extension of test range (3) reduction of test use paper.

1 はじめに

複写機をはじめとする情報機器は多機能、高性能化し、それを制御するプログラムも肥大、複雑化している。これに伴い機器が設計仕様通りに動作することを確認するテスト工程と、その中で検出された不具合の改修、確認を行うデバッグ工程での工数が著しく増大し、開発期間を圧迫する一要因となっている。

従来のテスト工程では、実際の複写動作を基にテストを行っているため、そのテストで要求される複写動作を実現することが難しい場合もある。例えばジャム（紙詰まり）やエラー処理のテストに至っては、その現象を発生させることが難しく、強いてはテスト用プログラムまでをも作り込まなければならないのが実状であった。

また、デバッグ工程における不具合の改修、確認方法においても、不具合発生時の複写モードで、その不具合が再現するまで複写動作を繰り返し行い、不具合の原因を解析していく手法が取られているが、発生頻度の少ない不具合では数千枚にもおよぶ複写動作が必要となる場合がある。さらには、不具合改修後の確認においても、同等枚数の複写動作が要求されるため、この作業にも多大な工数を要すると同時に、大量の紙が必要となり資源的な観点からも問題がある。

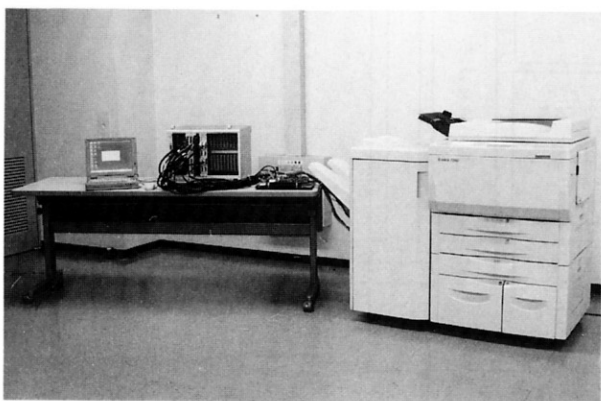


Fig.1 テストツールと複写機

今回、上記の問題点を改善すべく、プログラムのテスト工程とデバッグ工程を支援するツールを開発し、製品開発の中で試行したので紹介する。

2 情報機器制御システムの特徴

情報機器は機器に組み込まれたコンピュータシステム（組込みシステム）によって制御されており、次のような特徴がある。¹⁾

第一の特徴として、組込みシステムはセンサ信号をはじめとする制御対象からの信号を入力し、モータ等の制御対象へ制御命令を出力する、リアクティブ（反応型）システムであるうえ、規定の時間内に処理を行わなければならないリアルタイムシステムであることがあげられる。このためリアルタイムシステムの制御用プログラムのテストには、全ての制御対象が必要となる。²⁾

例えば複写機のフィニッシャーもリアルタイムシステムの一つである。このフィニッシャーの制御プログラムをテストする場合、複写機より送り込まれてくる紙の通過に伴うセンサ信号はもちろんのこと、複写機本体より送られてくるシリアル通信信号も制御対象となるので、テスト時にはフィニッシャー単体でテストを行うことができず、複写機本体またはそれに代わるテストツールが必要となる。

第二の特徴は、制御対象に予測、制御不可能な要素が含まれているために入力と出力の関係が一意に定まらないことである。

例えば複写機では、紙の特性や環境要因等により、給紙開始指令を出してから紙が通過検知センサに到達するまでの時間が複写動作の各紙毎に異なっていると共に、複写機内には複数の紙が存在しており、これらが複雑な関係を作りあげている。このため、テストで不具合が検出できてもその不具合が再現できず、デバッグを非常に困難なものにすると共に、不具合改修後の確認にも多大な工数を要する。不具合の再現性は、デバッグ工程において大きなウェイトを占めている。

* 機器開発統括部 第三開発センター

3 テストツール開発

3.1 目的

本ツール開発の第一の目的は、不具合発生時の状態を確実に再現させることにより、デバッグに必要な情報を提供すると同時に、不具合改修後の確認テストの工数を削減することにある。

第二の目的は、テスト対象となる機器に組み込まれたコンピュータシステム（以下、制御基板と記す）に対して、センサや他の制御基板からの通信信号といった制御対象信号の一部を本ツールから擬似的に与えることにより、実機では実現が難しいタイミング、例えば複数のセンサが同時に ON するといったタイミング等でのテストをも可能にし、ソフトウェアの品質向上に寄与することにある。

3.2 仕様

本ツールの仕様を Fig.2 に示す。

3.3 構成

今回開発したツールの構成を Fig.3 に示す。

ツールは、パーソナルコンピュータ、収集出力ユニット、信号変換ユニットより構成されている。

(1) パーソナルコンピュータ

パーソナルコンピュータでは、後述のデータ収集機能、シミュレーション機能を実行する際に必要となる各種の設定、および収集したデータの保存を行う。また、テストパターンに対応した設定をファイルで用意することにより、テストの度毎に個々の設定項目を手入力するといった作業を排除した。

〈デジタル入力〉	
チャンネル数：	
収集機能選択時	92ch (5V耐圧：72ch 50V耐圧：20ch)
シミュレーション機能選択時	53ch (5V耐圧：33ch 50V耐圧：20ch)
サンプリング周期	：1ms
〈デジタル出力〉	
チャンネル数：	39ch (シミュレーション時)
出力周期	：1ms
〈シリアル通信〉	
チャンネル数：	4ch (モニタ／出力：切替可)
ボーレート	：9600～102400 bps
〈周波数測定〉	
周波数範囲	：4Hz～5KHz
最大入力電圧	：5V

Fig.2 仕様

(2) 収集出力ユニット

収集出力ユニットは、収集出力 CPU ボード、デジタルデータの入力を行うデジタル入力ボード、デジタルデータの出力を行うデジタル出力ボード、シリアル通信モニタおよび擬似シリアル通信を行うシリアルボード、周波数測定を行うパルス測定ボードより構成されており、収集出力 CPU ボードがパーソナルコンピュータより送られてきた設定条件を基に各ボードを制御すると共に、信号変換ユニットから入力された各データは収集出力 CPU ボードを介してパーソナルコンピュータに送られる。

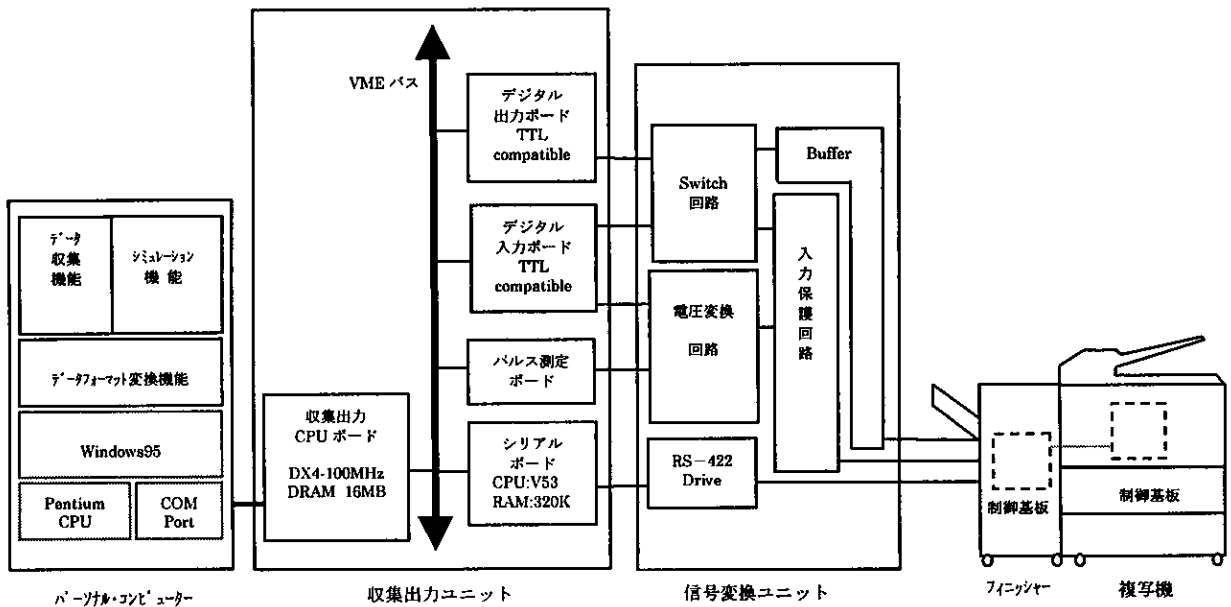


Fig.3 テストツール構成

3) 信号変換ユニット

信号変換ユニットは、高電圧のノイズから回路を保護する入力保護回路、制御基板からの高い電圧の信号を収集出力ユニットが入力可能な電圧に変換する電圧変換回路、制御基板に接続される制御対象信号として、実機の信号と収集出力ユニットからの擬似信号とを切り替えることができるスイッチ回路より構成されている。Fig.4 にスイッチ回路を示す。通常、スイッチはセンサ等の実機側に接続されており制御基板には実機の信号が入力されているが、シミュレーション機能によりシミュレーション動作が始まると同時に擬似信号側に接続され、制御基板には収集出力ユニットからの信号が入力されるように設計されている。

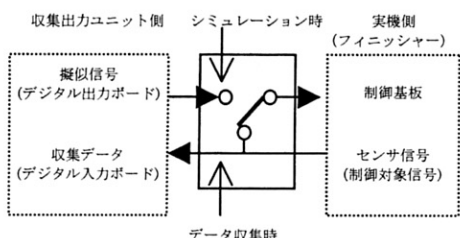


Fig.4 スイッチ回路

3.4 機能

(1) データ収集機能

データ収集機能はセンサ信号やモータ駆動制御信号等のデジタル信号、複写機の制御基板とフィニッシャーの制御基板とのシリアル通信信号、モータの速度を指示する周波数信号の各信号変化と、その時刻を同一の基準からの時刻として記録する機能である。

デジタル信号、シリアル通信信号、周波数信号をそれぞれ別々の測定器で計測する従来の測定方法では各信号間の時間的関係が不明確であり、制御対象のタイミング関係を把握することは困難であったが、本機能により各信号のタイミング関係が把握可能となり、テスト時の仕様確認が容易になる。

(2) シミュレーション機能

シミュレーション機能は、収集出力ユニットからのデジタル出力信号を特定のセンサ信号やシリアル通信信号の代わりに擬似信号として制御基板に与えると同時に、データ収集機能によるデータの収集を行う。また、擬似信号の対象となるセンサ等は個々に選択が可能である。Fig.5 にシミュレーション時の信号接続例を示す。ここでは、制御基板のセンサ信号 A がシミュレーションの擬似信号として選択されており、制御基板のセンサ信号 A には、フィニッシャー内に取り付けられているセンサ A からの信号に代わり、収集出力ユニットのデジタル出力信号が接続されている。また擬似信号の対象となっていないセンサ B の信号は、フィニッ

シャー内に取り付けられたセンサの信号が制御基板のセンサ信号 B に接続される。これと同時に、センサ信号 B およびモータの駆動制御信号は、データ収集機能によりその信号変化が記録される。この機能によりシミュレーションを行うと同時にその時の制御対象の挙動を把握できる。

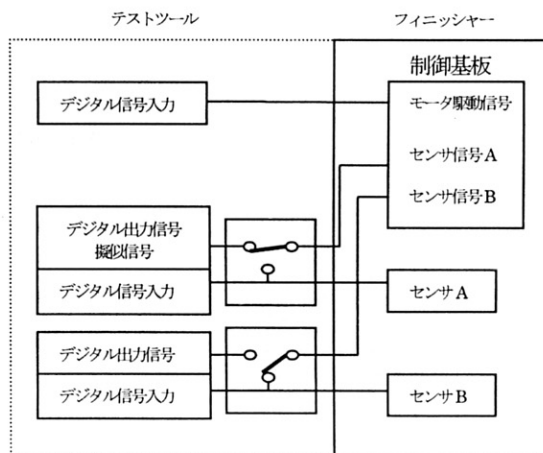


Fig.5 信号接続図

(3) データフォーマット変換機能

収集出力ユニット内部では、収集出力 CPU および各ボードがそれぞれ異なったフォーマットでデータを扱っている。本ツールでは、パーソナルコンピュータにこれらのフォーマットを市販の表計算ソフトや、データ表示ソフトといったアプリケーションソフトと互換可能なフォーマットに変換する機能をもたせた。これにより収集したデータの解析を容易にすると同時に、各アプリケーションソフトで加工したデータを本ツールのテストデータとして活用することができる。

Fig.6 に市販のタイミング表示ツールで表示したタイミングチャートを示す。

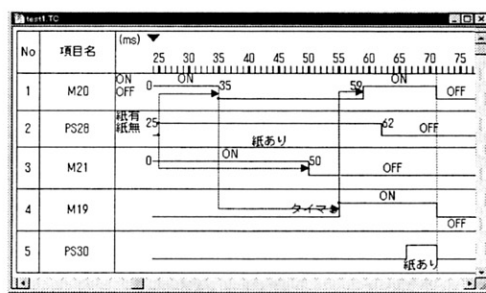


Fig.6 タイミングチャート表示例

4 複写機用フィニッシャーでの試行

4.1 テスト

今回、このテストツールを複写機用フィニッシャー制

御プログラムのジャム検知仕様確認テスト及び不具合検出後のデバッグに試行した。

テストは、本ツールとフィニッシャー単体で行い、ジャム検知仕様書に基づき、ジャム検知要因毎にジャムが発生するタイミング条件を作成し、そのタイミングでのシミュレーションを行った。また、実機での発生頻度が極めて少なく、実機を使った強制テストでは発生させることが難しい条件、例えば、設計上のタイミングが極めて近く、信号の発生順序が逆転が起り得るような場合については、その発生順序を逆にしたテストも行った。また、テストで検出された不具合の原因解析、改修後の確認テストにも、本ツールを使用した。

Fig.7 にテスト手順についてのフローチャートを示す。

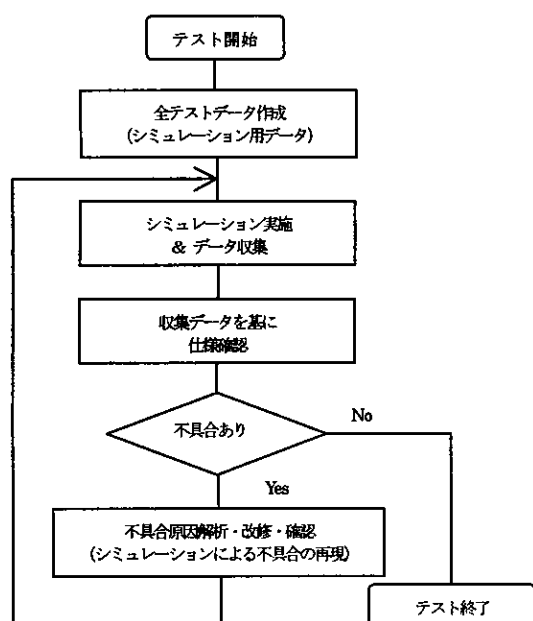


Fig.7 テスト手順

4.2 結果

今回10モードのテストを行った。テスト結果を Fig.8 に示す。

このテストによりジャム検知仕様を確認すると共に、2件の不具合を検出した。そのうち1件は本テスト実施前に数十万枚の通紙テストで一回だけ発生し未解決となっていた不具合であった。実機での再現テストを行ったが不具合が再現できず継続検討となっていたものである。しかし、今回本ツールを活用することにより、原因の特定、改修を行うことができた。

今回のテストでは、実際に紙を通して確認を行う従来の方法に比べ、仕様確認テストで約30%、不具合改修後の確認テストで約80%の工数を削減することができた。

	モード	原稿 枚数	部数 設定	ジャム発生条件		結果	ジャム コード
				部数	転写紙		
TEST1	ソト	2	2	2	2	不具合改修	10h
TEST2	ソト	2	2	2	2	不具合改修	10h
TEST3	ノット	1	5	3	—	OK	11h
TEST4	ステップ	2	3	3	1	OK	10h
TEST5	ステップ	2	3	3	1	OK	10h
TEST6	ステップ	3	2	2	3	OK	12h
TEST7	ステップ	3	2	2	—	OK	13h
TEST8	ステップ	3	2	1	—	OK	14h
TEST9	ステップ	2	3	1	—	OK	15h
TEST10	中折り	2	3	1	—	OK	16h

Fig.8 テスト結果

5 評価

本システムがプログラムのテスト工程とデバッグ工程において、以下の点で有効であることが検証できた。

5.1 再現性の確保

テスト時の動作状態を全て記録しているため、不具合発生時の状態を確実に再現させることが可能で、不具合解析に必要な情報を常に得ることができる。これにより、従来解析不可能になっていた不具合を改修できるようになった。

5.2 テストの強化

シミュレーション機能にてテスト時のタイミングを操作することにより、信号発生タイミングの逆転等、従来のテストでは行えなかったタイミング条件までテスト範囲を拡大し、予めプログラムの確認を行うことができた。

5.3 テスト紙の削減

本ツールを活用することにより、1枚も紙を使うことなく仕様確認を行うことができた。

6 むすび

複写機をはじめとする情報機器の制御ソフトウェア用テストツールを開発し、その有効性を検証できた。

今後は、フィニッシャー以外の機器への適用を図ると共に、無人シミュレーション、自動検証等の機能を拡充していきたいと考えている。

●参考文献

- 1) 高田広章, 情報処理, Vol.38 No.10, 870, (1997)
- 2) 内平直志, 川田秀司, 情報処理, Vol.39 No.1, 19, (1998)