

PLCopen の Safety ファンクションブロックを活用した安全プログラム自動化の実現

菅沼 拓・藤村 亮輔・久保 寛子・田熊 伸行

近年、製造業の現場においては、ニーズ多様化の時代に求められる「多品種変量」への対応が必要となっている。これに伴い、安全制御も柔軟に製造ラインの段取り替えが可能な安全システムの構築が必要であり、その解決策として、安全コントローラの導入が挙げられる。

標準化団体PLCopenによるSafetyファンクションブロック仕様の定義により、安全コントローラの導入障壁は徐々に下がってきてはいるものの、PLCopenのSafetyファンクションブロック仕様の柔軟性の高さゆえに、安全プログラムの未熟練者がこれらのファンクションブロックを使用することは容易ではない。また、将来の改良、拡張を考慮した安全システム構築に向けた、拡張性・可読性の高い安全プログラム作成も困難であるため、安全コントローラの導入障壁は依然高い状態であると言える。

そこで上記課題を解決するために安全プログラムの自動化を実現した。本稿では、オートメーションソフトウェア Sysmac Studio にて実現した安全プログラム自動化の技術成果とその効果について報告する。

その結果、Sysmac Studio で実現された「自動プログラミング」機能により、装置立ち上げ工数の短縮効果があることを確認した。

New automatic safety programming using safety function blocks of PLCopen

Hiromu Suganuma, Ryosuke Fujimura, Hiroko Kubo, Nobuyuki Takuma

In recent years, manufacturing companies need to meet the request of “multi-variety variable-volume production” in the era of diversified needs. Along with that, they also need to develop safety systems that can be flexibly modified according to the changeover of production lines. As a solution for that, safety controllers are often introduced.

Since PLCopen, the standardizing association defined the specification of the safety function blocks, introduction of safety controllers have gradually become less challenging. However, it is still not easy for unskilled developers in safety programs to use these function blocks, because the specification defined by PLCopen is highly flexible. Since it is also difficult to create a safety program with high scalability and readability considering future improvements and extensions, it can be said that the barriers to introduce safety controllers are still high.

To offer solutions for those issues, we realized automatic safety programming on the Omron's programming software Sysmac Studio. In this paper, we describe technical achievements and effects of automatic safety programming.

We confirmed that the “automatic programming” function of the Sysmac Studio can shorten development period of safety systems.

1. まえがき

1.1 安全コントローラ導入の必要性 近年、日本の製造業は、「人材不足の深刻化」という大きな環境変化に直面している。経済産業省が発行しているものづくり白書

2018¹⁾によると、94%の企業が人材確保に課題があり、3割超えの企業においてはビジネスにも影響があると回答している。その中で、特に確保に課題のある人材は「技能人材」であり、工場とその制御の変化に原因があると考えられる。

モノが不足していた時代の「少品種大量」工場から、ニーズが多様化する時代の「多品種変量」工場への変化対応により、製造ラインの柔軟な段取り替えが可能なシステム構築と保守が必要となっている。

安全制御においても同様に、柔軟に段取り替えが可能な安全システムの構築が必要になるが、従来のハードワイヤリングや安全リレーユニットでは配線の入替えや拡張のための変更にかかる時間がかかり、生産リードタイムを維持するための柔軟な安全システムを実現することができない。この対応策として安全コントローラ導入の必要性が高まっている。

安全コントローラとは、国際安全規格の認証を取得した安全制御のためのコントローラであり、安全回路のソフトウェア化や安全ネットワークによる省配線により、柔軟な安全システムを実現することを可能としている。

1.2 安全コントローラの現状 過去10年余り、IEC 61508²⁾を代表とする国際安全規格に対応した安全コントローラが各社から発売されている。また、「PLCアプリケーション開発の効率化」を目的とした標準化団体であるPLCopenは、トレーニングコストの低減、プログラム再利用性の向上、安全規格認証コストの低減などの効果を狙い、プログラムに関する共通定義化を推進している。

その結果、安全プログラムで使用するファンクションブロックやデータ型が共通化され、プログラム作成のハードルが下がり、安全コントローラの導入障壁は低くなってきている。

オムロンもPLCopenのSafety ファンクションブロックに対応した安全コントローラNX-SLシリーズとオートメーションソフトウェアSysmac Studioをリリースし、ニーズ多様化に対応できる安全システムを提供している。

2. 課題と対策

2.1 PLCopen の Safety ファンクションブロックの課題 PLCopenのSafety ファンクションブロック仕様は、複雑な安全システムであっても柔軟に対応できるように定義されている。そのため、安全プログラムの設計者は、これらのファンクションブロックに定義されている多くのパラメータの中から、必要なパラメータを選択し、適切な値を設定する必要がある。例えば、インターロック用のリセット信号だけで3つのパラメータが存在し、それぞれに設定が必要となる。

加えて、PLCopenのSafety ファンクションブロックで定義されている安全入力用のファンクションブロックは、国際安全規格ISO13849-1³⁾で要求されているパルスリセット信号OFF→ON→OFFによるスタート/リスタートインターロックをサポートしていない。そのため、複数のファンクションブロックの組み合わせと適切なパラメータの設定により、規格要求に対応したプログラムを設計者自らが作成する必要がある。

上記のとおり、PLCopenのSafety ファンクションブロックを使用した安全プログラムの作成には国際安全規格に基づいた安全プログラムの知識と経験が必要であると言える。

2.2 安全プログラムの課題 また、進化する「多品種変量」工場への変化対応として、将来の改良、拡張に耐える安全システムの構築を実現する必要がある。例えば拡張のためにI/O点数を増やす場合、それに伴うプログラムの追加量が必要以上に多くなったり、また追加プログラムによってプログラムが複雑化し、可読性が低くなってしまったりすることがある。こういった事態を避けるために、あらかじめI/O点数の増減を考慮した拡張性と可読性の高い安全プログラムを実現することが、変化対応への課題であるといえる。

2.3 対策 前述した課題を解決すべく、安全コントローラのプログラム設計を行うオートメーションソフトウェアSysmac Studioにおいて、プログラミングレスの簡単設定のみで自動的にプログラムを生成する機能を実現した。

従来、パルスリセット信号によるスタート/リスタートインターロックの実現や、I/O点数の増減に対応可能な拡張性と可読性の高いプログラムの作成には熟練設計者の技能が必要であり、プログラミングツールによる安全プログラム生成の自動化は困難であった。

今回、オートメーションソフトウェアSysmac Studioに安全プログラムのノウハウを自動プログラミング機能として組み込むことで、設計者が設定する項目を最小限に抑え、また入力処理部、出力処理部、論理演算部の3つに分割された拡張性と可読性の高い安全プログラムを容易に作成できるようになり、未熟練設計者であっても熟練設計者と同等の安全プログラムの作成を可能とする。

3. 自動プログラミング機能の概要

自動プログラミング機能は、表1に示す設定項目をインプットすることで、自動的にプログラムを生成する機能である。

表1 自動プログラミングの設定項目

設定項目	内容
基本設定	安全プログラムのリセット信号として使用する変数を指定する。
入力設定	安全入力に割り付けられた変数とリセット種別、および使用するセーフティファンクションブロックの名称を指定する。
出力設定	安全出力に割り付けられた変数と、外部機器モニタリング (EDM) 使用の有無を指定する。
期待値設定	各入力変数に対する出力変数の期待値 (0または1) をマトリックスで設定する。

自動プログラミング機能で生成されるプログラムは、図1のようになる。

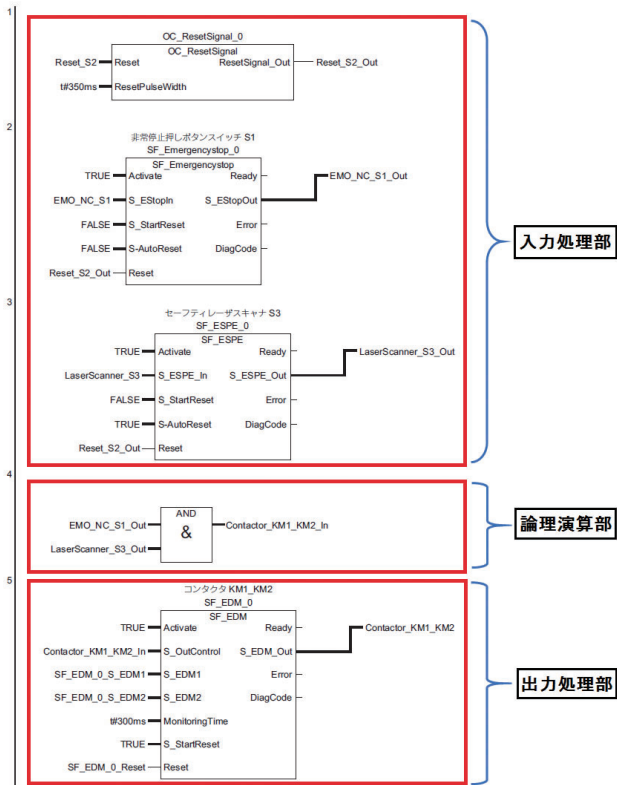


図1 自動プログラミングで生成されるプログラム

生成されるプログラムは、安全スイッチや安全センサからの入力を処理する「入力処理部」、セーフティリレーへの出力を処理する「出力処理部」、それらを接続する「論理演算部」の3つのブロックからなり、これら3つのブロックは内部変数を介して接続される。ここで、内部変数とは、プログラム構成単位内のみで使用されるローカル変数のことである。

次章では、自動プログラミング機能の実現方法について、ブロックごとに順を追って説明する。

4. 自動プログラミング機能の実現方法

4.1 入力処理部の生成 入力処理部では、入力信号に割り付けられた変数を取り込み、PLCopen の Safety ファンクションブロックで定義された安全入力用ファンクションブロックを使用してスタート／リスタートインターロックなどを行う。

しかし、PLCopen の Safety ファンクションブロックの安全入力用ファンクションブロックはパルスリセット信号によるスタート／リスタートインターロックに対応していない。プログラムで実現するには、信号のタイミングを考慮し、複数のファンクションブロックを組み合わせるロジックの構築が必要である。

そこで我々はパルスリセット信号に対応したユーザ定義ファンクションブロック「OC_ResetSignal」を開発した。ユーザ定義ファンクションブロックとは、複数のファンクションブロックを組み合わせたものを部品化し、ユーザプ

ログラムで使うことができるファンクションブロックのことである。図2にユーザ定義ファンクションブロック「OC_ResetSignal」の外観を示す。

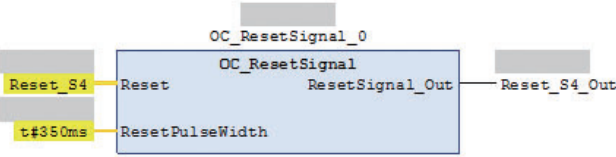


図2 ユーザ定義ファンクションブロック「OC_ResetSignal」

「OC_ResetSignal」の内部ロジックは非公開としているが、Sysmac Studioの自動プログラミング機能では、このユーザ定義ファンクションブロックと、PLCopen の Safety ファンクションブロックを自動的に組み合わせることで、スタート／リスタートインターロックを実現している。

また、安全入力用ファンクションブロックは多くのパラメータと設定のバリエーションを持つため、パラメータを選択して適切な値を入力することが設計者に求められる。

そこでファンクションブロックのパラメータ設定を容易にするために、後述するパラメータ分析を行い、設定が必要なパラメータの絞り込みを行った。安全入力用ファンクションブロックに必要なパラメータは、表2にある3つに絞り込むことができた。

表2 安全入力用ファンクションブロックに必要なパラメータ

パラメータ	内容
安全入力	安全スイッチ/安全センサなどの入力信号
Auto Reset	リスタートインターロックの有効/無効を切り替えるパラメータ
Reset	リスタートインターロック有効時のリセット信号

出力パラメータと入力パラメータを次の観点で分析し、必要なパラメータを絞り込んだ。

- ・観点
 - 設計者による設定が必要
 - 設定は必要だが、自動設定することで、設計者による設定は不要
 - 設定は不要

まず、出力パラメータを分析した結果を表3に示す。

表3 出力パラメータの分析結果

パラメータ	分析内容	分析結果
安全制御に関連しないパラメータ	エラーフラグやエラーコードなどのモニタ用途の出力パラメータ。ファンクションブロックの動作や安全制御に関連しないため、設定しない。	設定不要
安全出力	安全制御信号として論理演算部と接続する。自動で内部変数を割り当てる。	自動設定

続いて、入力パラメータを分析した結果を表4に示す。

表4 入力パラメータの分析結果

パラメータ	分析内容	分析結果
一般的な用途において使用しない入力パラメータ	ファンクションブロックの動作開始条件フラグなどのパラメータ。 自動で定数を割り当てる。	自動設定
安全入力	安全スイッチ／安全センサなどの入力信号。 設計者が変数を割りつける。	設定対象
Auto Reset	リスタートインターロックの有効／無効を切り替えるパラメータ。 設計者が設定によって切り替える。	設定対象
Reset	リスタートインターロック有効時のリセット信号。 設計者が変数を割りつける。	設定対象

上記分析によって絞り込んだ3つのパラメータと、安全入力用のファンクションブロックの種別を、Sysmac Studioの自動プログラミングの入力処理部の設定として表5のように定義した。

表5 自動プログラミング機能の設定項目(入力処理部)

設定項目	関連パラメータ	内容
リセット信号	Reset	リスタートインターロック有効時のリセット信号に割り当てる変数
変数	安全入力	安全スイッチ／安全センサなどの入力信号に割り当てる変数
リセット種別	Auto Reset	Manualリセット (リスタートインターロック機能有効) または Autoリセットの選択
ファンクションブロック名	—	安全入力用のファンクションブロックの種別

Sysmac Studioの入力処理部のパラメータ設定画面を図3に示す。

▼ 基本設定				
リセット信号 Reset_S4				
▼ 入力設定				
No.	変数	コメント	リセット種別	ファンクションブロック名
1	EMO_NC_S1		Manual	SF_EmergencyStop
2	EMO_NC_S2		Manual	SF_EmergencyStop
3	LaserScanner_S3		Manual	SF_ESPE

図3 入力処理部パラメータ設定画面

上記設定項目から生成される入力処理部のプログラムは図4のようになる。

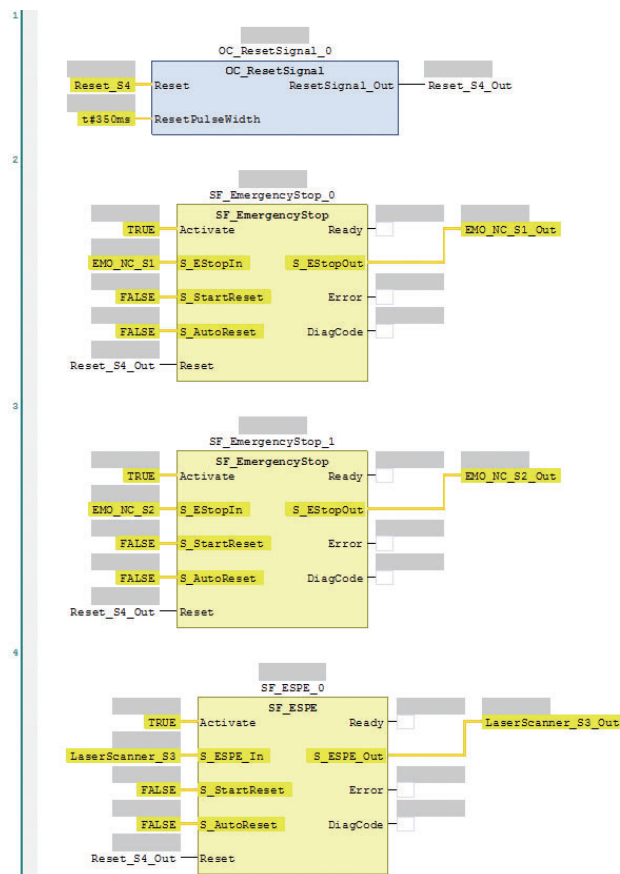


図4 生成される入力処理部のプログラム

各安全入力用ファンクションブロックからの安全信号出力には内部変数を設定し、後述の論理演算部との接続に使用する。また、パルスリセット信号に対応したOC_ResetSignalをプログラムの先頭に配置し、その出力パラメータには内部変数を設定する。その内部変数を各安全入力用ファンクションブロックのReset入力パラメータに設定することでパルスリセット信号によるスタート／リスタートインターロックを実現している。

4.2 出力処理部の生成 出力処理部では、出力信号に割り付けられた変数に対して安全制御信号を出力し、外部機器モニタリング (EDM) を行う。

入力処理部と同様、ファンクションブロックのパラメータ設定を容易にするために、後述するパラメータ分析を行い、設定が必要なパラメータの絞り込みを行った。安全出力用ファンクションブロックに必要なパラメータは表6にある1つに絞り込むことができた。

表6 安全出力ファンクションブロックに必要なパラメータ

パラメータ	内容
安全出力	動力供給を制御する安全リレーの出力信号

出力処理部においても、入力処理部と同じ観点で分析することで、必要なパラメータを絞り込んだ。

まず、出力パラメータを分析した結果を表7に示す。

表7 出力パラメータの分析結果

パラメータ	分析内容	分析結果
安全制御に関連しないパラメータ	エラーフラグやエラーコードなどのモニタ用途の出力パラメータ。ファンクションブロックの動作や安全制御に関連しないため、設定しない。	設定不要
安全出力	動力供給を制御する安全リレーの出力信号。設計者が変数を割りつける。	設定対象

続いて、入力パラメータを分析した結果を表8に示す。

表8 入力パラメータの分析結果

パラメータ	分析内容	分析結果
一般的な用途において使用しない入力パラメータ	ファンクションブロックの動作開始条件フラグなどのパラメータ。自動で定数を割り当てる。	自動設定
安全入力	安全制御信号として論理演算部と接続する。自動で内部変数を割り当てる。	自動設定
EDM入力	外部機器モニタリング信号。設計者が後から変更する。自動で内部変数を割り当てる。	自動設定
EDMリセット	EDMのエラーを取り除くパラメータ。設計者が後から変更する。自動で内部変数を割り当てる。	自動設定

上記分析によって絞り込んだ1つのパラメータと、安全出力用のファンクションブロック SF_EDM の使用有無を、Sysmac Studio の自動プログラミングの出力処理部の設定として表9のように定義した。

表9 自動プログラミング機能の設定項目(出力処理部)

設定項目	関連パラメータ	内容
変数	安全出力	動力供給を制御する安全リレーの出力信号に割り当てる変数
EDM使用	—	外部機器のモニタリングを行うためのファンクションブロック「SF_EDM」使用有無の指定

Sysmac Studio の出力処理部のパラメータ設定画面を図5に示す。

▼ 出力設定			
No.	変数	コメント	EDM使用
1	Contactor_KM1_KM2		TRUE
2	Contactor_KM3_KM4		TRUE

図5 出力処理部パラメータ設定画面

上記設定項目から生成される出力処理部のプログラムは図6のようになる。

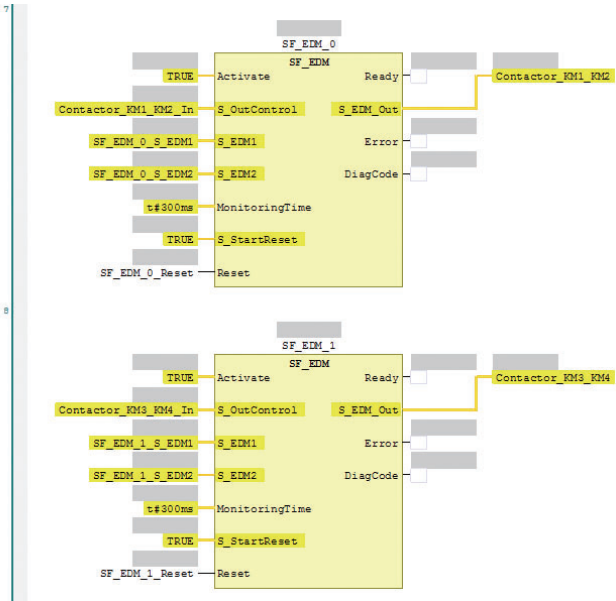


図6 生成される出力処理部のプログラム

安全制御信号である SF_EDM ファンクションブロックの入力パラメータ S_OutControl には内部変数を設定し、後述の論理演算部との接続に使用する。外部機器モニタリング (EDM) を使用しない場合は、SF_EDM ファンクションブロックを介さず、論理演算部において直接指定した変数へ出力する。

4.3 論理演算部の生成 論理演算部は、入力処理部と出力処理部を接続する論理回路である。今回我々は、安全入力を OFF にしたときの安全出力の期待値を記述したテーブルを元に、論理回路を生成する技術を開発した。

安全制御信号に対する論理演算は、入力処理部と出力処理部による論理積 (AND ファンクション) で構成される。これを、安全設計者が保有している設計ドキュメントを活用し、自動で論理演算部の生成を行う。

安全設計者が保有している設計ドキュメントである、安全入力と安全出力の関係性を表すテーブルの例を表10に示す。

表10 安全入力と安全出力の関係を表すテーブルの例

		安全出力		
		コンタクタ KM1_KM2	コンタクタ KM3_KM4	...
安全入力	非常停止スイッチS1	OFF	OFF	...
	非常停止スイッチS2	OFF	ON	...
	レーザスキャナS3	ON	OFF	...
	:	:	:	

OFF: 遮断する、ON: 遮断しない

このテーブルの情報を元に設定される Sysmac Studio の自動プログラミング機能の期待値設定画面を、図7に示す。

▼ 期待値設定					
No.	変数	コメント	リセット種別	Contactor_KM1_KM2	Contactor_KM3_KM4
1	EMO_NC_S1		Manual	0	0
2	EMO_NC_S2		Manual	0	1
3	LaserScanner_S3		Manual	1	0

図7 自動プログラミングの期待値設定画面

この期待値設定から、それぞれの安全出力に対応する安全入力を抽出し、AND ファンクションで結ぶことで論理演算部は以下の図8のように実現される。

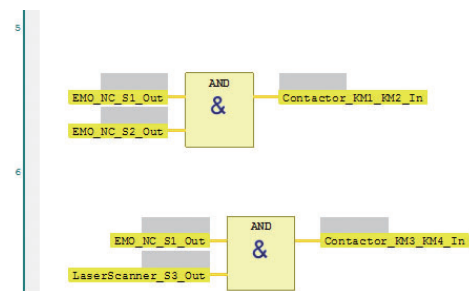


図8 生成される論理演算部のプログラム

今回の簡単な構成においては、設計者によるプログラム作成で誤りなく論理演算回路を実現することは可能だが、複雑な構成になると多大な単純作業とそれ故のケアレスミスが発生してしまうため、この自動化が有効であると言える。

4.4 論理演算部を入力処理部と出力処理部から分離するメリット 「入力処理部」、「出力処理部」、「論理演算部」に分けて、自動的にプログラムを生成する方法について述べたが、これら3つのブロックに分離することで、プログラムの拡張性と可読性が高まると言える。

例えば、1つの安全出力に対して複数の安全入力が影響する一般的な安全システム事例において、論理演算部を分離する場合のプログラムを図9に、論理演算部を分離しない場合のプログラムを図10に示す。

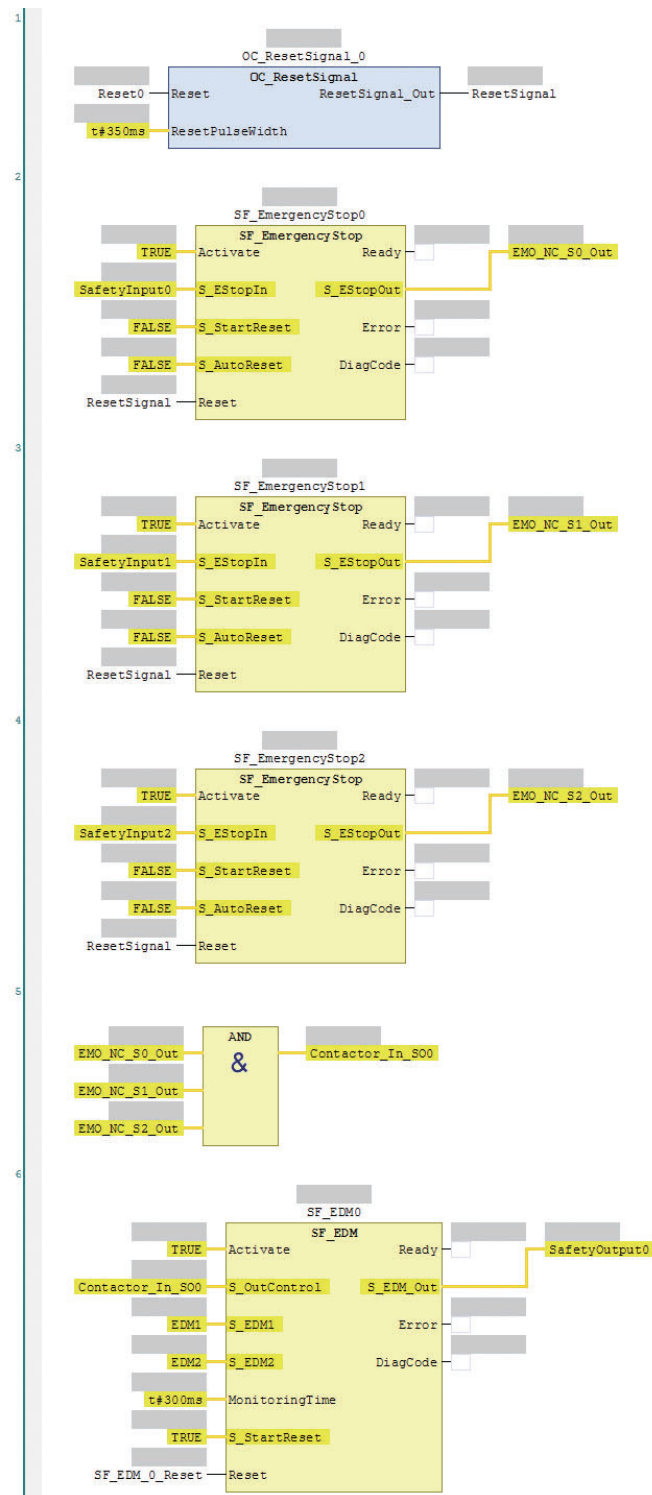


図9 論理演算部を分離するプログラムの例

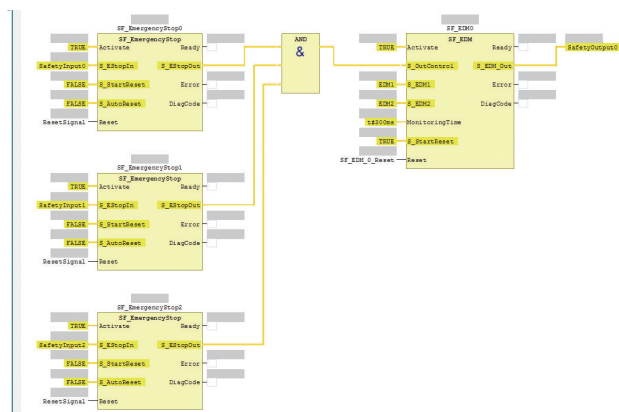


図10 論理演算部を分離しないプログラムの例

これらのプログラムに安全出力1点を追加した場合に追加が必要なプログラムを図11と図12に示す。

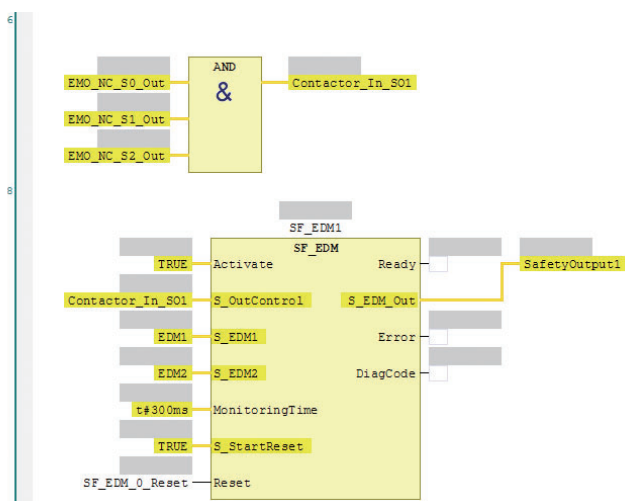


図11 論理演算部を分離する場合の追加されるプログラム

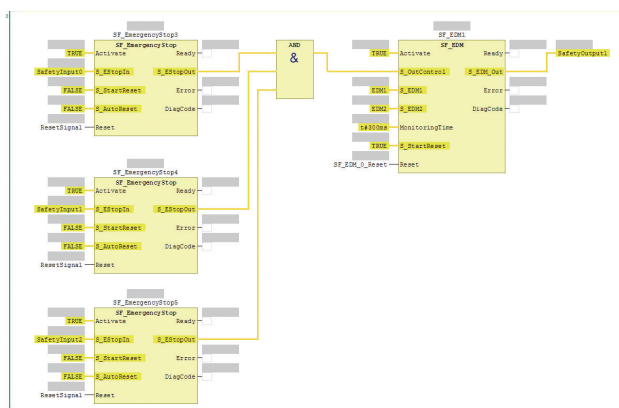


図12 論理演算部を分離しない場合の追加されるプログラム

論理演算部を分離する場合においては、図11に示すとおり論理演算部の追加と安全出力用のファンクションブロックの追加のみで可能である。

一方で論理演算部を分離しない場合においては、図12に示すとおり、プログラム全体を複製し、安全入力用のファンクションブロックの設定を行い、安全出力用のファンク

ションブロックを追加する必要がある。

よって、内部変数を使用して論理演算部を分離する方が、追加されるプログラムが簡単で、かつ変更量が少ないことから、拡張性と可読性が高まると言える。

4.5 効果の検証 今回Sysmac Studioに機能追加した自動プログラミング機能によって、必要最小限の簡単な設定で、拡張性と可読性の高いプログラムを自動生成することを実現した。

さらに、Sysmac Studioの自動プログラミング機能を使用した場合の導入効果について、自社内で妥当性検証を行った。その結果、図13に示すように、ある安全システムのモデルケースにおいてプログラム作成に要する時間が90%短縮され、その結果、装置設計全体工数の40%を短縮する効果があることを確認した。

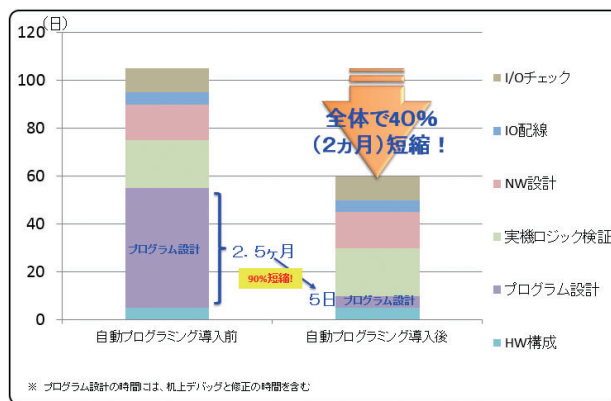


図13 自動プログラミング機能導入の効果

5. むすび

本稿では、PLCopenのSafetyファンクションブロックの活用により、安全プログラム自動化の技術革新について述べた。

必要パラメータの絞り込み技術と論理回路生成技術により、必要最小限の設定項目から熟練設計者と同レベルの安全プログラムを自動生成することで、装置立ち上げ工数の短縮効果があることを確認した。また、論理演算部を分離することでI/O点数の増減に対応可能な拡張性と可読性の高い安全プログラムを定義し、自動プログラミング機能に組み込むことで、柔軟な安全システムの構築を可能とした。

今後においては、市場からのフィードバックを受け、自動プログラミング機能を進化させていくとともに、さらに多様化が進むお客様のニーズに対応できる技術革新への取り組みを継続する。

最後に、Sysmac Studioの自動プログラミング機能の実現にあたり、多大なご協力をいただいた、事業企画・コンサルティング・販売の方々をはじめ、開発に携わった方々に深く感謝申し上げます。

参考文献

- 1) 経済産業省:「2018年版ものづくり白書 - 第2節 人手不足が進む中での生産性向上の実現に向け、「現場力」を再構築する「経営力」の重要性」
http://www.meti.go.jp/report/whitepaper/mono/2018/honbun_pdf/pdf/honbun01_01_02.pdf
- 2) IEC 61508:2010 Functional safety of electrical/electronic/programmable electronic safety-related systems
- 3) ISO 13849-1:2015 Safety of machinery -- Safety-related parts of control systems -- Part 1: General principles for design

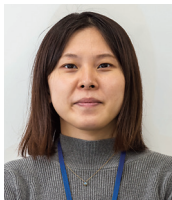
執筆者紹介



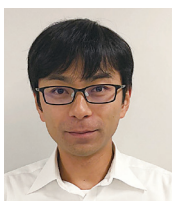
菅沼 拓 Hiromu Suganuma
インダストリアルオートメーションビジネス
カンパニー 商品事業本部
コントローラ事業部 ソフトウェア開発部
専門: ソフトウェア工学



藤村 亮輔 Ryosuke Fujimura
インダストリアルオートメーションビジネス
カンパニー 商品事業本部
コントローラ事業部 ソフトウェア開発部
専門: ソフトウェア工学



久保 寛子 Hiroko Kubo
インダストリアルオートメーションビジネス
カンパニー 商品事業本部
コントローラ事業部 ソフトウェア開発部
専門: ソフトウェア工学



田熊 伸行 Nobuyuki Takuma
インダストリアルオートメーションビジネス
カンパニー 商品事業本部
コントローラ事業部 コントローラPMG
専門: ソフトウェア工学