

COM を含む MFC による配電制御 システムの構築

沖縄職業能力開発大学校 林 文 彬
石 川 功
職業能力開発総合大学校 八 田 昌 之

The Development of Local Distribution System Control Program by MFC and COM
Bunhin LIN, Isaō ISHIKAWA, Masayuki HATTA

要約 ソフトウェア産業において、Windows を中心とするアプリケーションの開発はなくてはならない存在である。本論文はオブジェクト指向プログラミングの機能を持つ Visual C++ を基に、Win32API、COM (Component Object Model) と MFC (Microsoft Foundation Class) を使い、配電制御システムを構築する。本システムを構築する目的は新規応用短期課程 (企業人スクール) のコースを開発するために、教育訓練に供する教材に用いる一つの具体的な対象事例として活用することである。構築したシステム事例には C++ で構築したものがあつたが¹⁾、従来のシステムの実行環境は DOS 環境で、キーボードによる操作・入力であるので、システムの実用化にはまだ改良の余地が残つてある。新しいシステムの構築には Win32API、COM および MFC などの Windows プログラミングを導入してシステムを完成したので、実行環境はグラフィック画面化で、操作方法もマウスのクリックだけで実行できる。これによって、システムの実用化はさらに大きな一歩が進んだ。また、構築したシステムを教材とする応用短期課程 (企業人スクール) の新規コースは今年度 (平成14年度) に開発され、試行コースの開講に対して受講生が定員満員に集まつた。アプリケーションシステムの GUI 化のニーズが高まりになつた現在、本システムによる応用短期課程 (企業人スクール) のコース開発は、業界の要望に合うという確認ができた。

はじめに

WindowsOS の普及に従つて、Windows 環境を持つアプリケーションシステムの開発はソフトウェア産業において大切な存在になりつつある。本論文は業界の時代ニーズに合わせて、新規応用短期課程 (企業人スクール) のコースを開発するため、教育訓練に使われる教材に用いる一つの具体的な対象システム事例を構築した。

構築したシステム事例はオブジェクト指向の機能を持つ Visual C++ を基に、WinAPI32、COM (Component Object Model) および MFC (Microsoft

Foundation Class) などの新しい Windows プログラミングを導入し、配電系統停電時における復旧操作の自動化である。この対象事例はかつてエキスパートシステムの構築手法を導入して C++ 言語によるシステムを構築することがあつたが¹⁾、システムの実行環境は DOS 環境で、操作方法もキーボードにより行われる。操作員の系統操作を補助するとき、実行の利便性は十分とは言えない。

それで、本論文はシステムの GUI 化のソフトウェア開発技術を導入してシステムを構築した。これにより、配電制御システムの実用化が図られると共に、システムの構築ノウハウを応用短期課程 (企業人スクー

ル)のコース開発に活用することが求められる。

システム構築の内容にはまず、実会社の配電システムを対象システムモデルとする。この会社の系統に各種の故障形態を予め想定し、もし停電事故が発生すると、どうやって復旧操作を行うかを考える。そして、検討の結果でまとめた各故障パターンの復旧操作手法を知識ベースとしてコンピュータに蓄え、エキスパートシステムの推論手法を導入してシステムを構築する。

問題記述¹⁾

近年、配電機器の性能の向上と共に機器の信頼性も高くなってきた。こういう給電信頼性の高い配電系統に、企業または工場内の系統事故による停電件数は少なくなり、停電時の復旧操作を経験した操作員もそれなりに減っている現状である。

一方、経営者は企業経営コストを節約するため、コ・ジェネレーションシステムなどの効率高い分散電源が積極的に導入されつつある。しかし、企業内の配電系統は複数の自家用発電機を電力会社の系統と連系して負荷に電気を供給すると、系統構成が複雑化となり、管理運営も難しくなる。もし系統外部または内部に何らかの事故により停電が発生する場合、停電操作経験の少なさと停電に伴う心理的要因に加えて操作員

が適切な復旧操作を行うのは難しいと予想される。

そこで、本論文は実規模の企業内配電システムを対象システムモデルとする。この系統に各種の故障形態を予め想定し、もし停電事故が発生する場合、どうやって復旧操作を行うかを考える。方法としては、まず、系統の操作則とベテラン操作員の経験則を参考にして、停電の故障パターンに対応できる緊急停電操作手順および故障除去後の復電操作手順を一々検討する。検討の結果でまとめた各故障パターンの操作手法を知識ベースとしてコンピュータに蓄えてシステムを構築する。

1 対象系統

図1は本研究に対象とする配電系統である。左上のガスタービン発電機と燃料電池の2つのコ・ジェネレーションシステムが整備され、その出力は電力会社の系統と連系して保安、ビルA系、B系およびプラントなどの負荷に電気を供給する。

図1の太線は平常時に使われて負荷に電気を供給する。細線は予備線として平常時に使われていないが、太線が故障のときに利用される。また、コ・ジェネレーションシステムは24時間の運転に対して、非常用発電機は平常時に運転していない。

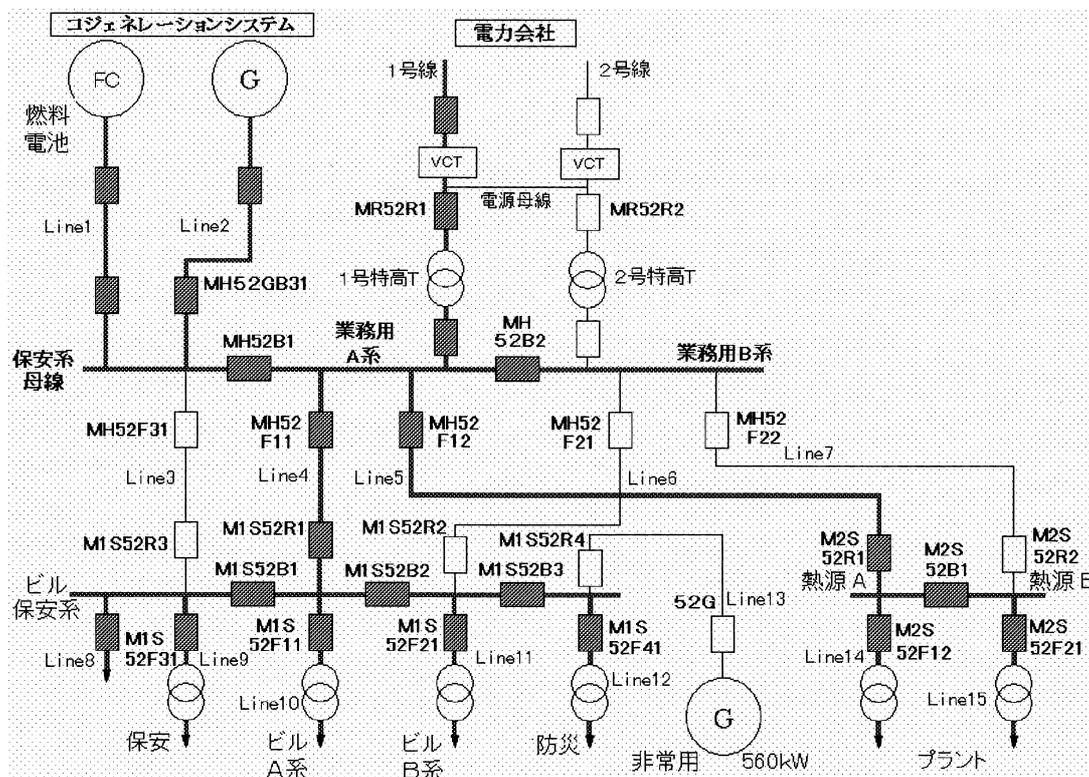


図1 対象系統

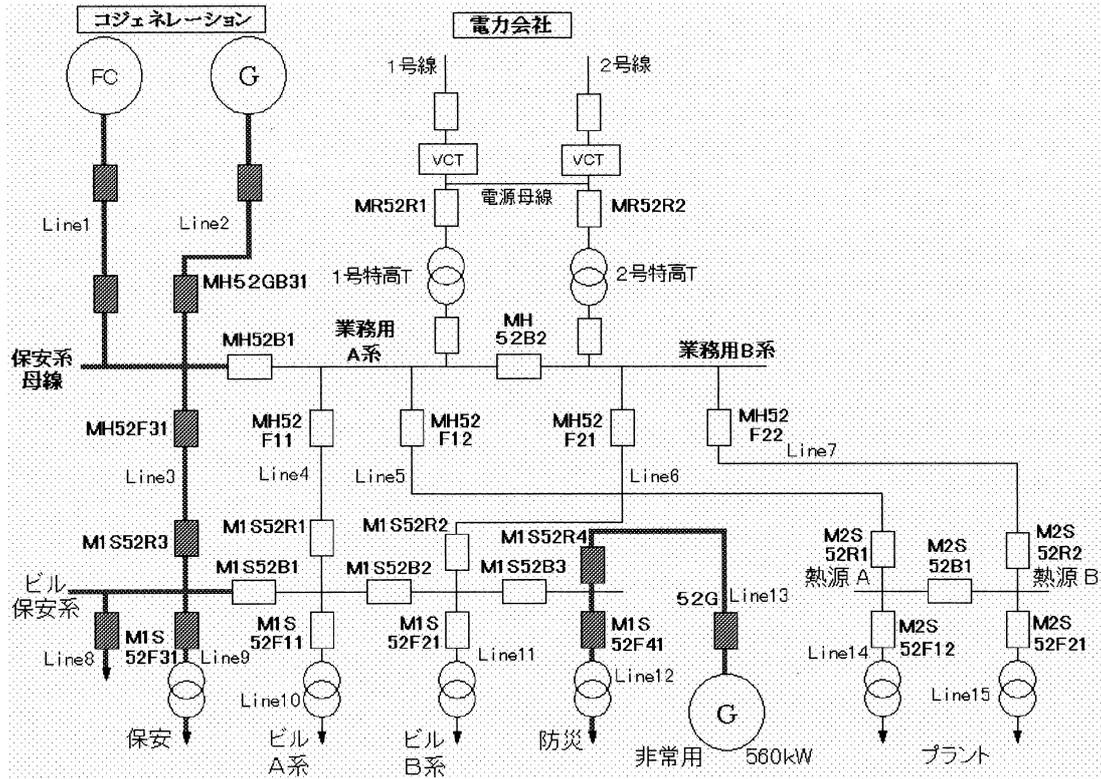


図2 特高停電

2 事故時の緊急操作例

電力系統は事故により、停電が発生することは滅多にないが、僅かながら外的要因による事故の発生の可能性はある。本システムは系統に事故が発生する可能性のあるパターンを想定して復旧手順を推論する。例えば、電力会社の系統は停電が発生する場合、まず、コ・ジェネレーションシステムの運転に影響を与えないため、遮断器 MH52B 1 を遮断して、電力会社系統との連系を解列する。その後、系統の操作則に基づいて、コ・ジェネレーションシステムの出力は保安系の負荷だけに電気を供給する。このとき、非常用発電機も起動・発電により防災系に電気を供給する。

ここまで特高停電の緊急処理手順が終了し、電気供給範囲は図2の示すような2つブロックとなる。

以上の例の示すように、本システムでは対象系統にいろんな停電パターンが発生する場合を予め想定する。それらの停電パターンの復旧操作はベテラン操作員の経験を参考し、それぞれの停電パターンに対応する復旧操作手順を検討する。さらに検討した操作手順をルール化すると共に、エキスパートシステムを構築する一方法としてすでに発表された¹⁾。

COMを含むMFCによるシステムの構築

1 オブジェクト指向データベース構築

本システムでは現実の系統事例をコンピュータシステムの世界へ置き換えるため、オブジェクト指向という概念に基づいた手法を使ってデータベースを構築した。

具体的に、対象配電系統のデータを配電線、母線および遮断器の3つの項目に分類してクラス化した。各クラスに関連する系統要素(データ)がデータメンバーとして定義される。また、クラス内のデータメンバーをアクセスするためのメンバー関数(プログラム)が用意され、データメンバーと一体化して系統の情報を管理する。表1は各クラスに定義されているデータメンバーとメンバー関数を示す。

表1から、本システムは各クラスの構造に基づいてクラスのオブジェクトを宣言したり、メンバー関数に經由してデータメンバーをアクセスしたりすることができる。これによって、システムは常に系統の停電・受電の最新情報を把握することと系統の充・受電操作ができるようになる。また、図3は各クラスのオブジェクトの一例を示す。実際にシステムの構築にあたって、クラスとオブジェクトの宣言はそれぞれ

表1 クラスの定義

クラス名	データメンバー	メンバー関数
line (配電線)	char name [20]; bus* busfrom ; bus* busto ; char breaker 1 [10], Breaker 2 [10]; int charge ;	line() ; line (char *na,bus& bus1 ,bus& bus2 , char *c,char *d,int f); void inputcharge (int p); void sijyuuden (int p); void jyuden (int p); int charged() ;
bus (母線)	char name [20]; int charge ;	bus() ; bus (int a,char *p); void inputcharge (int p); int charged() ;
breaker (遮断器)	char name [20]; line* linen ; int charge ;	breaker (int a,char *p,line& q); void inputcharge (int p); void sijyuuden (int p); void jyuden (int p); int charged() ;

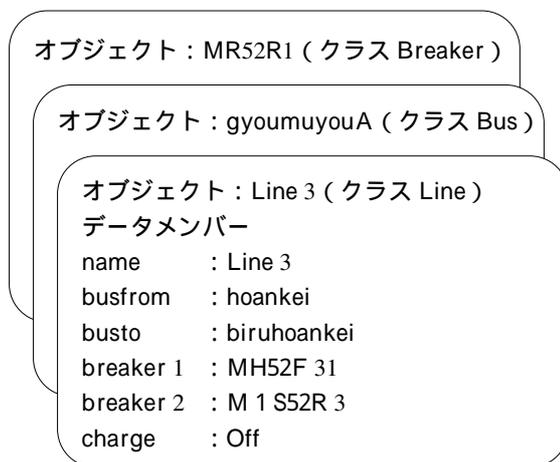


図3 各クラスのオブジェクト宣言

「HaidenView.h」と「HaidenView.cpp」で行われる。

2 クラス間の包含関係

系統の復旧は充電操作または受電操作を行う必要がある。充電操作は図4の示すように遮断器 MH52F12 が閉じることによって「業務用 A 系母線」が受電になる。同じように、受電操作は遮断器 M 2 S52R 1 が閉じることで、「熱源 A 母線」が受電になる (図5 に参照)。

しかし、図4 5のような充・受電操作をプログラムで実行する場合、まずクラス「Breaker」のオブジェクトからクラス「Line」のオブジェクト、さらに「Line」のオブジェクトからクラス「bus」のオブジェクトを呼び出して充・受電操作を実行する。データの隠蔽性を持つオブジェクト指向プログラミング C++

においてはオブジェクト「line 5」のデータメンバー「busto」に「熱源 A 母線」という名前を設定しても、オブジェクト line 5 からオブジェクト「熱源 A 母線」のデータを読み出せない。それで、図6の示すようにオブジェクトの宣言には「ポインタ」、データの代入時には「参照」という手法を使うことで、クラス「Breaker」のメンバー関数「jyuden」からクラス「Line」のメンバー関数「jyuden」に経由してクラス「bus」のメンバー関数「inputcharge」を操作できるようになった。

3 システムの構築

本論文は Visual C++ の AppWizard 機能によりプログラムの自動生成を行って、GUI 化のシステムを開発する。また、選択用ボタンをサポートするダイア

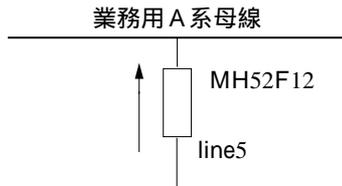


図 4 充電操作

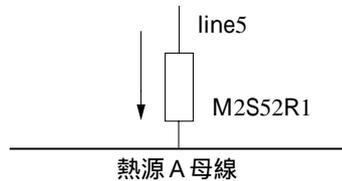


図 5 受電操作

```
class line : public CHaidenView {charge=f ;
    char name[20] ;
    bus* busfrom ; //クラス bus を指す point
    bus* busto ; //クラス bus を指す point
    char sw 1 [10], sw 2 [10] ;
    int charge ;
public:
    line(char *na,bus& bus 1,bus& bus 2,
        char *c,char *d,int f )
    { strcpy( name, na ) ;
      busfrom=&bus1 ; //参照代入
      busto=&bus2 ; ; //参照代入
      strcpy( sw1 , c ) ;
      strcpy( sw 2 , d ) ;
```

図 6 クラス間の包含関係

ログボックスを扱うための専用クラスの定義は Class-Wizard を利用して作成する。

構築したシステムには直接にコードの追加や変更を施したファイルとその役割が表 2 の示すようになる。また、特高停電操作を行う時に上記ファイルと内

部の関数の流れは図 7 の示すようになる。

最初に、メニューバーの停電操作がクリックされると、図 7 の STEP 1 の ChaidenView クラスの関数 OnSeigyobar() が起動される。

OnSeigyobar() から、クラス CseigyoDlg のダイアログボックスを呼び出し、「特高停電」、「ビル A 系母線故障」および「ビル B 系母線故障」の 3 つのボタンが表示される (STEP 2)。これらのボタンはそれぞれ CseigyoDlg クラスの関数 OnTokukouteider()、OnAkeikosyo()、OnBkeikosyo() と関連しているので、どちらのボタンが選ばれるとそれに対応する関数が起動される。例えば「特高停電」のボタンが選ばれる場合、関数 OnTokukouteider() が起動され、その結果が関数 OnSeigyobar() に伝わる (STEP 3)。そして、関数 OnSeigyobar() は「特高停電」のルールグループを実行して復旧操作を行う。その後、実行は STEP 4 に移って、ChaidenView クラスの関数 OnKairetsuhandar() が起動される。

OnKairetsuhandar() は Ckairetsuhandan クラスのダイアログボックスを呼び出すことによって、「解列成功」、「解列失敗」という選択ボタンが出てくるので、操作員の判断を待っている。

もし「解列成功」というボタンが選択されれば、Ckairetsuhandan クラスの関数 OnKairetsuseikou() が起動されて、結果を関数 OnKairetsuhandar() に伝え、OnKairetsuhandar() はこれに基づいて復旧操作は続く (STEP 6)。しかし、「解列失敗」というボタンが選択されるならば、OnKairetsuhandar() はコージェネレーションシステムの再起動を指示してから、再び STEP 5 に戻る。これによって、特高停電操作の流れができた。

表 2 主なファイルと役割

ファイル名	役割
HaidenView. cpp	オブジェクトの定義、 メニューバーのメニュー実行 ダイアログボックスの起動とボタン選択後の実行
MainFrm. cpp	Window を希望する通りの大きさに定義
Doukitounyu. cpp	同期投入成功か否かの選択用ダイアログボックスを定義
Kairetsuhandan. cpp	解列成功か否かの選択用ダイアログボックスを定義
KosyouDlg. cpp	故障の種類を選択するためのダイアログボックスを定義
SeigyouDlg. cpp	全停電か部分停電などの選択用ダイアログボックスを定義
HaidenView. h	配電線、母線、遮断器のクラス定義

ファイル : HaidenView.cpp	ダイアログボックス
STEP1 ,メニューバーの停電操作をクリック ChaidenView :: OnSeigyobar()起動	STEP2 ,停電形態選択 CseigyoDlg :: OnTokukouteider() CseigyoDlg :: OnAkeikosyou() CseigyoDlg :: OnBkeikosyou()を選択 (ファイル : SeigyoDlg.cpp)
STEP3 ,「停電形態選択」の結果を受け取る ChaidenView :: OnSeigyobar()	
STEP4 ,解列成功か ChaidenView :: OnKairetsuhandar()	STEP5 ,解列成功か否か選択 Ckairetsuhandan :: OnKairetsuseikou() Ckairetsuhandan :: OnKairetsusippa()を選択 (ファイル : kairetsuhandan.cpp)
STEP6 , 「解列成功か」の結果を受け取る ChaidenView :: OnKairetsuhandar()	

図7 特高停電操作時に関数操作の流れ

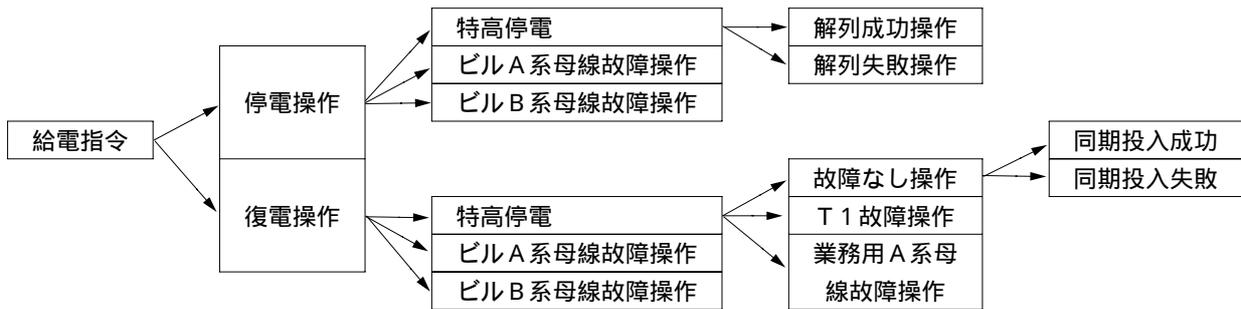


図8 給電指令操作の流れ

実行の結果

1 実行の流れ

本システムにおける復旧操作は自動的に行う操作と給電指令を要する操作がある。復旧時、自動的に行う自動操作の進行に従っていくつかの操作は操作員の判断により給電指令を出す必要がある。図8は復旧操作時、操作員の判断が必要とする給電指令である。復旧操作は停電操作と復電操作の2つ部分からなる。処理の初期段階は給電指令の停電操作と復旧処理のどちらかの選択から始まる。その後、操作員が現実の系統の状況に照合しながら、判断を下したり、システムが自動操作に基づいて復旧を行ったりする。このようにして、系統の復旧は自動操作と給電指令による操作の2つが交互に進行しながら、操作が進む。

最後に、復旧操作は終わると、システムは再度「給電指令」のモードに戻って次の指令待になる。

2 システムの実行

本システムの実行はメニューバーの「停電操作」と「復旧処理」の選択から始まる。ここで「停電操作」の「特高停電」を例にして実行を行う。

まず、メニューバーの「停電操作」をクリックして実行する(図9)。「停電操作」のクリックにより、図



図9 停電操作開始

10の「特高停電」、「ビルA系母線故障」、「ビルB系母線故障」の3つの選択ボタンを持つダイアログボックスが表示される。ここで、「特高停電」を選択する。

システムは自動操作により遮断器 MH52B 1 を切っ、て、系統の解列操作を行う。その後、図11のような「解列成功」、「解列失敗」の指令選択ダイアログボックス

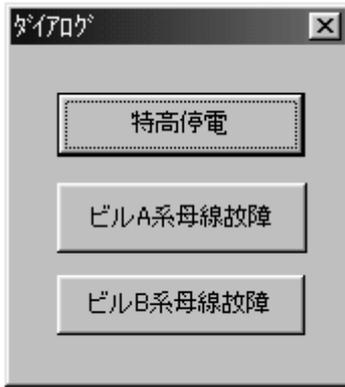


図10 停電パターンの選択



図13 特高停電故障パターン



図11 解列成功か否かの確認



図14 同期投入が成功したかの確認

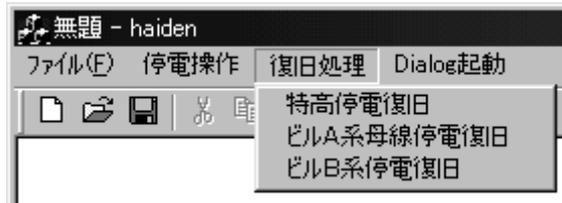


図12 復旧処理の選択

が出てきて、操作員の確認に求める。

もし「解列失敗」を選択する場合、システムはコ・ジェネレーションシステムの再起動を指示し、コ・ジェネレーションシステムの再起動操作を行う。

しかし、「解列成功」を選択する場合、システムは順次ビル A 系、ビル B 系、プラントの負荷を系統と切り離すような自動操作指令を出す。最後に、復旧手順は非常用発電機を起動・発電して防災系に電気を供給する。ここまで、操作が終了する。このとき系統の受電範囲は図 2 の示すようになる。

次に復電操作の場合は、図 2 のような状態から、特高 1 号線に電気があれば、復電操作に入る。はじめに図 12 のメニューバー「復旧処理」がクリックされれば、「特高停電復旧」、「ビル A 系母線停電復旧」、「ビル B 系母線停電復旧」の 3 つサブメニューを選択することができる。ここで「特高停電復旧」を選択する。

この選択によって、図 13 の「故障なし」、「T 故障」、「A 系故障」の選択ボタンが現れるので、操作員は特高停電が起こった原因に基づいて判断する。もし「故障なし」を選択する場合、「故障なし」の復旧操作が始まる。復旧の終了段階で、復旧される系統はコ・ジェネレーションシステム電源と特高電源の 2 つブロックになり、2 つブロック間の同期投入の条件（周波数と電圧が一致）が揃えば、遮断器 MH52B 1 を投入し、2 つブロックの系統を連系させる。また、同期投入の成功か否かは操作員の確認に求めるこ

同期投入成功

保安系母線とビル保安系母線と切り離す (MH52F31, M1S52R3 OFF)

ビル A 系母線とビル保安系母線と併用 (M1S52R1 ON)

特高停電復旧終了

図15 同期投入成功の場合

とがある(図14)。復旧は図15の示す段階で終了する。

結論

本システムは、職業能力開発総合大でまとめた応用短期課程(企業人スクール)におけるコース開発のための教材開発によるものである¹³⁾。ここで、その結果をまとめる。

1. Win32API, COM および MFC を使った Windows プログラミングの導入により、システムの実行環境はグラフィック画面化になり、見た目がおしゃれと共に、操作環境もマウスのクリックだけで操作方法が簡単で直感的である。これにより利用者に優しい仕事環境ができて、配電制御システムの実用化に向けてさらに一歩進んだ。
2. 本システムは応用短期課程(企業人スクール)の新規コース開発に教育訓練用の教材として開発されたものである。今年度(平成14年)沖縄職業能力開発大学校で試行コース「Windows プログラミングによるアプリケーションシステムの開発(Visual C++ 篇)」が開講される。開講に際して、受講者が定員満員に集まることになった。これにより、本システムの構築を事例とする応用短期課程コースは企業のニーズに合い、システムを構築する目的の応用短期課程コース開発は達成したとということである。

今後、システムから出した給電指令は文字でなくて、システムのスケルトンの形で画面に表示できるプログラムを改良したいと思う。

[参考文献]

- (1) 林文彬・他：「配電系統停電時における復旧操作のエキスパートシステム化」、職業能力開発報文誌、第14巻第1号 pp.1~8、(2002年)
- (2) 専門委員会：「電力システムのエキスパートシステム化」、電気学会技術報告(部)339号、1990
- (3) N. Kakimoto, B. P. Lin, M. Hayashi: "Restoration of Power System from Complete Outage by Expert System", Third Symposium on Expert Systems Application to Power System (p.262 - 269) (1991)
- (4) 垣本・林：「超高压系統全停電時の系統再構成における給電指令の自動選択」、電気学会論文誌 B, 113巻 6号, PP613 - 620 (平5)
- (5) David J. Kruglinshi 著、榊・梅原共訳：「Inside

Visual C++ Version 4」, アスキー出版局

- (6) 桜田・田口著：「Visual C++ 4.0 プログラミング入門」, アスキー出版局
- (7) ハーバート・シルト著、柏原訳：「C/C++ プログラマのための Windows95 プログラミング」, shoeisha
- (8) 豊田著：「Visual C++ プログラマのための COM 入門」, shoeisha
- (9) 加藤著：「ATL による COM プログラミング入門」, 技術評論社
- (10) 大園著：「最新&次世代データベースの基本と仕組み」, 秀和システム
- (11) J.G. ヒューズ著、植村監訳：「オブジェクト指向データベース」, サイエンス社
- (12) 林・石川・八田：「COM を含む MFC による配電制御システムの構築」, 第10回職業能力開発研究発表講演会予稿集(2002年)
- (13) 林文彬：「COM を含む MFC による配電制御システムの構築」, 職業能力開発総合大学校、応用研究論文(2002年)