制御 PC 電源瞬断ソリューション(INtime)

株式会社マイクロネット
http://www.mnc.co.jp

TEL: +81(0)299-90-1733
FAX: +81(0)299-92-8557

REV.1 2020/07/30 RELEASE

目次

本書で使用するマークについて

	ノート: 操作方法や手順等の補足情報や注釈を説明しています。
	情報: 製品を利用する上で有効な豆知識となる説明をしています。
	警告: 製品仕様上注意が必要な事象について説明しています。

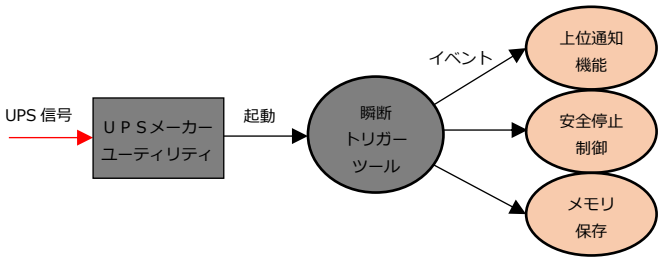
1　概要

本ドキュメントは、電源瞬断により制御 PC を正しく終了処理を行えない事で制御機器の危険動作やデータ破損などの対策方法について記述しています。

本ドキュメントをお読みいただく前に「制御 PC 電源瞬断ソリューション(共通)」ドキュメントをお読みください

2　瞬断イベント処理

電源瞬断により UPS ベンダーが提供している「UPS ユーティリティ」から弊社の「瞬断トリガーツール」起動する事により各処理にイベントを発行する事が出来ます。



ユーザーが作成したセマフォまたは瞬断トリガーツールと一緒に配布しているライブラリの API(WaitPowerFailureEvent())を使用して瞬断トリガーツールからのイベントを待ちます。

イベントを受けた制御アプリケーションは電源瞬断に必要なイベント処理を行ってください。

通知するセマフォの INtime Node 名やカタログ名は、瞬断トリガーツールの起動パラメータで変更する事が出来ます。

例：瞬断イベント処理　（※制御内容はシステムよって異なります）

```
// 瞬断イベント時のアクション
void PowerFailureHandler()
{
  // 瞬断イベント通知の待機
  WaitPowerFailureEvent();

  // 制御対象を安全状態にする
  GoToSafe();
  // メモリデータの保存
  SaveMemoryToDisk();
  // 上位サーバへの通知
  ReportToServer();
}

// 瞬断イベント通知待ち
void WaitPowerFailureEvent()
{
  RTHANDLE hSem,hRoot;
  hRoot=GetRtThreadHandles(ROOT_PROCESS);
  hSem=CreateRtSemaphore(0,1,0);
  CatalogRtHandle(hRoot,hSem,"PFevent");
  WaitForRtSemaphore(hSem,1,WAIT_FOREVER);
}
```

瞬断トリガーツールから
イベント通知受ける

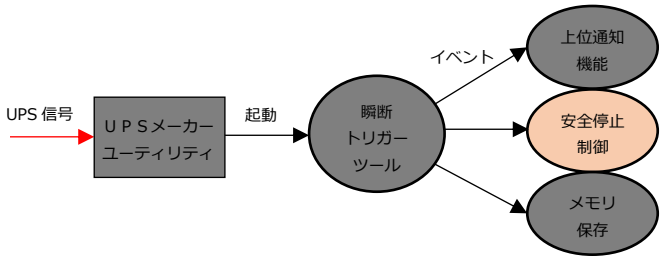
瞬断トリガーツールからのイベント通知を受けた時は瞬断対処を行います。

処理はユーザーがシステムに合わせ作成します。

3　制御対象を安全状態にする

電源瞬断に必要なリアクションンの 1 つとして「安全状態にする」があります。

瞬断により UPS から信号を受け制御対象を安全状態（動作を停止させる、釣り上げている物を下す、ブレーキをかけておくなど）にする必要があります。



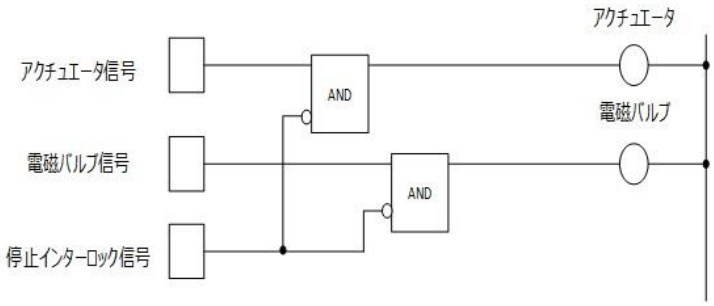
例：安全状態にする

1.制御信号出力

1）アクチュエータ信号の OFF

2）電磁バルブ信号の OFF

3）停止インターロック信号の ON



2.モーター制御

1）安全位置に移動させる

2）電磁ブレーキをかけて保持する

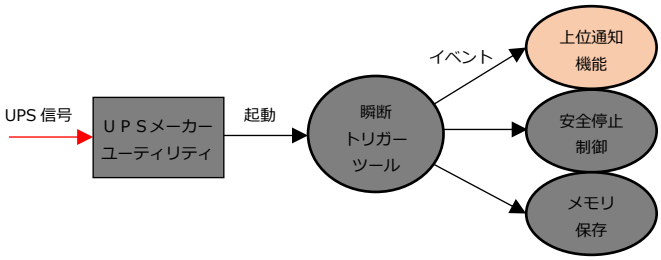
プログラミング例（※制御内容はシステムよって異なります）

```
// 制御対象を安全状態にする
void GoToSafe()
{
  // アクチュエータ制御信号を OFF
  outbyte(actuator,0x00);

  // 電磁バルブ制御信号を OFF
  outbyte(valve,0x00);

  // 停止インターロック信号を ON
  outbyte(safetylock,0x01);
}
```

4　上位システムに通知する対処

電源瞬断に必要なリアクションンの 1 つとして「上位システムに通知」があります。

瞬断により UPS より信号を受け制御 PC をシャットダウンする事を上位システムに通知が必要です。



プログラミング例（※制御内容はシステムよって異なります）

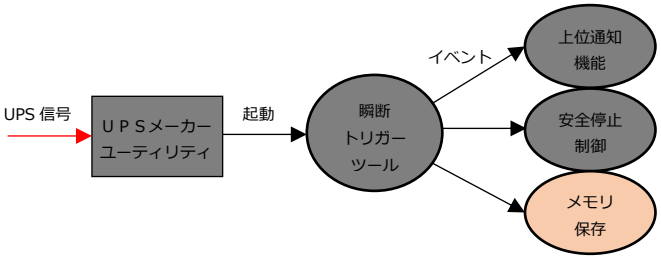
```
// 信号の処理ハンドラ
void ReportToServer()
{
  // UDP でサーバに送信する
  SOCKET s;
  s=socket(AF_INET,SO_DGRAM);
  sendto(datas,sizeof(datas),to);
  soclose(s):
}
```

- 通信回線を通じて報告
- TCP,UDP,RS232C 等
- socketAPI の利用

5　メモリ上のデータを退避する対処

電源瞬断に必要なリアクションンの 1 つとして「メモリ保存」があります。

メモリ内のみに保持しているデータは制御 PC シャットダウン時に消えてしまう為、保持が必要なメモリ内のデータをファイルに出力します。



プログラミング例（※制御内容はシステムよって異なります）

```
   // 信号の処理ハンドラ
void SaveMemoryToDisk()
{
    // C 関数 fwrite で書き出す
     FILE *fp;
    fp = fopen(“save.bin”,”wb”);
    fwrite( datas,sizeof(datas),1,fp);
    fclose(fp);
}
```

- データをファイル保存する
- C 関数 fwrite が利用可能

6 改訂履歴

版数	発行日	改定内容
第 1 版	2020 年 7 月	初版発行