

EmbedWare/SysMon Ver 7.1 for Windows

説明書

このページは空白です。

ごあいさつ

このたびは、「EmbedWare/SysMon® Ver 7.1」をご利用いただき、誠にありがとうございます。
本書は、ハードウェアの動作状態を監視する EmbedWare/SysMon® Ver 7.1 のインストール方法や、
操作方法などについて記述しています。本書をよくお読みになって正しくお使いください。また、
本書を大切に保管していただきますようお願いいたします。
本書が、EmbedWare/SysMon® Ver 7.1 を活用していただくために、みなさまのお役に立つことを
願っております。

2019年11月

EmbedWare、EmbedWare/SysMon、EmbedWare/SysMon Entry は、株式会社 PFU の日本国における
登録商標です。

その他の会社名、製品名などは、各社の商標または、登録商標です。

Microsoft Corporation のガイドラインに従って画面写真を使用しています。

© PFU Limited 2018-2019

●本書の構成について

第1部 *EmbedWare/SysMon Entry*

第1章 概要

EmbedWare/SysMon Entry の概要について説明します。

第2章 インストールとアンインストール

EmbedWare/SysMon Entry のインストール／アンインストール方法について説明します。

第3章 *EmbedWare/SysMon Entry を利用する*

EmbedWare/SysMon Entry の利用方法について説明します。

第2部 *EmbedWare/SysMon SDK*

第1章 概要

EmbedWare/SysMon SDK の概要について説明します。

第2章 インストールとアンインストール

EmbedWare/SysMon SDK のインストール／アンインストール方法について説明します。

第3章 API 仕様

EmbedWare/SysMon SDK を利用するにあたって、必要な API 仕様について説明します。

第4章 API 使用方法

EmbedWare/SysMon SDK の API 使用法について説明します。

付録

EmbedWare/SysMon Entry の通知メッセージ、GUI メッセージ、EmbedWare/SysMon SDK API ディスク情報取得、およびシステムエラー監視の仕組みについて説明します。

●本書の表記について

本書で使用している記号とその意味を以下に示します。

 注意	お使いになるときに注意していただきたいことや、してはいけないことを記述しています。 必ずお読みください。
 備考	操作に関する補足事項を記述しています。 必要に応じてお読みください。

●略称

本書では、以下の用語について省略して表記する場合がありますのでご了承ください。

正式名称	略称
Windows Server® 2008 R2	Windows Server 2008 R2
Windows® 7 Professional Operating System	Windows 7 Professional、 Windows 7
Windows® Embedded Standard 7	Windows Embedded Standard 7
Windows® 10 IoT Enterprise	Windows 10 IoT Enterprise、 Windows 10
EmbedWare/SysMon®	EmbedWare/SysMon
AW400 モデル 020A ソフトウェア RAID	ソフトウェア RAID

●本書に掲載している画面について

本書では、主に Windows 7 の画面を例として説明します。

目次

第1部 EmbedWare/SysMon Entry

第1章 概要

1.1	EmbedWare/SysMon Entry とは.....	3
1.2	機能.....	3
1.3	動作環境.....	5

第2章 インストールとアンインストール

2.1	インストール	7
2.2	アンインストール	13
2.3	版数確認.....	15

第3章 EmbedWare/SysMon Entry を利用する

3.1	温度監視.....	19
(1)	温度の表示.....	19
(2)	監視条件の設定.....	20
(3)	しきい値の設定.....	21
3.2	電圧監視.....	22
(1)	電圧の表示.....	22
(2)	監視条件の設定.....	23
3.3	ファン監視	25
(1)	ファン回転数の表示	25
(2)	監視条件の設定.....	26
(3)	しきい値の設定.....	27
3.4	S.M.A.R.T. 監視	28
(1)	S.M.A.R.T. ステータスの表示	28
(2)	監視設定	29
(3)	S.M.A.R.T. アトリビュート情報の表示	30
3.5	RAID 監視	31
(1)	RAID 情報の表示	31
(2)	監視設定	32
3.6	SVRx RAID 監視	33
(1)	SVRx RAID 情報の表示	33
(2)	監視設定	34

3.7	ソフトウェア RAID 監視	35
(1)	ソフトウェア RAID 情報の表示	35
(2)	監視設定	36
3.8	システムエラー監視	37
(1)	システムエラー情報の表示	37
(2)	監視設定	38
3.9	システム情報表示	39
3.10	パフォーマンス情報	40
(1)	パフォーマンス情報の表示	40
(2)	監視設定	40
3.11	EmbedWare/SysMon イベントログ表示	41
3.12	BIOS イベントログ表示	42
3.13	BMC SEL 表示	43
3.14	ウォッチドッグタイム設定	44
3.15	オプション設定	46

第 2 部 EmbedWare/SysMon SDK

第 1 章 概要

1.1	EmbedWare/SysMon SDK とは	51
1.2	機能	51
1.3	アプリケーション開発環境	52
(1)	インストール環境	52
(2)	アプリケーション作成時の留意事項	52
1.4	アプリケーション実行環境	53
(1)	インストール環境	53
(2)	アプリケーション実行環境作成時の留意事項	53

第 2 章 インストールとアンインストール

2.1	インストール	55
2.2	アンインストール	72
2.3	版数確認	76

第 3 章 API 仕様

3.1	ESM_Open() 関数	77
3.2	ESM_Close() 関数	78
3.3	ESM_Get() 関数	79
3.4	ESM_GetSmart() 関数	85

3.5	ESM_RaidInfo() 関数	86
3.6	ESM_SvRAIDInfo() 関数	88
3.7	ESM_QueryError() 関数	89
3.8	ESM_GetWDTTimeOut() 関数	90
3.9	ESM_SwitchWDT() 関数	91
3.10	ESM_WDTHeartbeat() 関数	92
3.11	ESM_ReadGPI() 関数	92
3.12	ESM_ReadGPO() 関数	93
3.13	ESM_WriteGPO() 関数	93
3.14	ESM_GetCpuCount() 関数	94
3.15	ESM_GetCpuPerform() 関数	94
3.16	ESM_GetSmartEx() 関数	95
3.17	ESM_GetSvDiskType() 関数	96
3.18	ESM_GetSvDiskSmartAttr() 関数	97
3.19	ESM_GetSwRAIDCount() 関数	98
3.20	ESM_GetSwRAIDInfo() 関数	99
3.21	ESM_GetSwDiskType() 関数	101
3.22	ESM_GetSwDiskSmartAttr() 関数	102

第 4 章 API 使用方法

4.1	API マルチスレッド対応	103
4.2	API 使用例	104

付録

付録 A	通知メッセージ一覧	105
付録 B	GUI メッセージ一覧	111
付録 C	EmbedWare/SysMon SDK API ディスク情報取得について	112
付録 D	システムエラー監視の仕組みについて	113

第1部 EmbedWare/ SysMon Entry

この章では、EmbedWare/SysMon Entry の概要について説明します。

1.1 EmbedWare/SysMon Entry とは

コンピューターのハードウェアが正常な動作状態にあるかを監視するソフトウェアです。EmbedWare/SysMon Entry は、コンピューターの状態を常時監視し、異常を検出した場合に、コンピューターの利用者に通知します。

1.2 機能

EmbedWare/SysMon Entry には、以下の機能があります。

機能	説明
状態監視	コンピューターの温度、電圧値、ファン回転数の現在の値を表示します。これらの値には、上限または下限値があらかじめ設定されており、異常の有無を容易に確認できます。
S.M.A.R.T. 監視	内蔵ディスクデバイスの S.M.A.R.T. (Self-Monitoring, Analysis and Reporting Technology) 機能による異常検出の有無を表示します。
RAID 監視	チップセット内蔵 RAID コントローラー、RAID カード (タイプ 1)、RAID カード (タイプ 2)、ソフトウェア RAID の状態を表示します。
システムエラー監視	BIOS イベントログの記録に基づいて、PCI バス、メモリ、および BMC エラーの有無を表示します。
パフォーマンス監視	CPU の実周波数情報を表示します。
異常通知	状態監視、ディスク監視、システムエラー監視で異常を検出した場合、ビープ音を鳴らし、検出した異常の内容を示すメッセージを表示します。
ログ記録、表示	状態監視、ディスク監視、システムエラー監視で異常を検出した場合、EmbedWare/SysMon のイベントログファイルおよび Windows イベントログに記録します。また、ログ表示機能では、EmbedWare/SysMon イベントログ、BIOS イベントログ、および BMC SEL (Baseboard Management Controller System Event Log) を表示します。EmbedWare/SysMon イベントログは EmbedWare/SysMon Entry で独自に記録しているログです。異常を検出した場合にメッセージを記録します。
シャットダウン処理	状態監視で異常を検出した場合、コンピューターの電源を切断します。
トレース機能	状態監視の履歴をトレースファイルに記録します。

機能	説明
過去最高値・過去最低値の表示	状態監視で記録した、過去最高値・過去最低値を表示します。
ウォッチ ドッグタイマ	EmbedWare/SysMon Entry のサービスプログラムが、一定時間動作できない状態になった場合、事前に指定されているアクションを実施します。 ほかのアプリケーションでウォッチ ドックタイマを制御する場合は、EmbedWare/SysMon SDK を使用してください。その場合、EmbedWare/SysMon Entry のウォッチ ドッグタイマ設定をオフにしてください。
システム情報表示	コンピューターの BIOS 版数、総通電時間を表示します。
BMC SEL 制御	BMC SEL とは、BMC 内部で保持されるシステムイベントログです。BMC SEL のクリア実行および、記録サイズが最大値に達した場合の動作を設定します。



- EmbedWare/SysMon Entry は、監視対象項目のエラーを検出しても、装置のアラーム ランプを点灯しません。
 - BMC に関する機能は BMC 搭載機種で使用可能です。
 - RAID 監視で監視対象としている RAID カードは、2 種類存在します。
 - RAID カード（タイプ 1）は PCIe インターフェイスカードタイプです。
 - RAID カード（タイプ 2）は SATA インターフェイスカードタイプです。
- 機種によって搭載される RAID カードは異なります。
搭載 RAID カードが不明な場合は、担当営業にご相談ください。

1.3 動作環境

EmbedWare/SysMon Entry の動作環境を以下に示します。

OS	: Windows 7 Professional SP1 以降、Windows 10 IoT Enterprise
更新プログラム	: 標準の Windows7 および Windows Server 2008 R2 では、SHA-256 証明書をサポートしていません。EmbedWare/SysMon Entry のドライバは SHA-256 証明書により署名されています。Microsoft 社の更新プログラム KB3033929 を適用するか、またはこれを含む更新プログラムを適用してください。
ディスク資源	: 約 9MB ~ 約 71MB
メモリ資源	: 監視サービス 約 7MB ~ 約 22MB GUI 約 50MB ~ 約 100MB

BMC 搭載の有無	OS	IPMI ドライバ	開始必須サービス
BMC 搭載機種	● Windows 10 ● Windows Server 2008 R2 以降	必須 (IPMI ドライバは、OS インストール時に標準インストールされます)	● Remote Procedure Call(RPC) ● Windows Management Instrumentation(WMI)
	● Windows 7 ● Windows Embedded Standard 7	不要	無
BMC 非搭載機	OS 問わず	不要	無



「システムの復元」が有効になっている場合は、「システムの復元」の設定に従い、ディスク資源が使用されます。



- SMBus (System Management Bus) にアクセスするアプリケーションや Windows 7 で BMC (Baseboard Management Controller) にアクセスするアプリケーションなどがある場合、動作保証されません。
- BMC 搭載機種で IPMI ドライバが必須の OS の場合、OS 標準の "Microsoft 汎用 IPMI 準拠デバイス" を無効化またはアンインストールしないでください。無効またはアンインストールした場合、動作保証されません。
- ウオッチドッグタイマ設定を行う場合は、本製品以外にウォッチドッグタイマへのアクセスを行っていないことを確認してください。本製品以外にアクセスがある場合、動作保証されません。
- インストールに失敗した場合、プログラムをインストールしたフォルダーおよびその配下にファイルが残る場合があります。再インストールしない場合は、手動で削除してください。
- 上記以外の環境をご使用のお客様は、担当営業にご相談ください。

第2章 インストールとアンインストール

この章では、EmbedWare/SysMon Entry のインストールとアンインストールの方法について説明します。

2.1 インストール

◇インストールする前に

EmbedWare/SysMon Entry をインストールする前に、以下のことを確認してください。

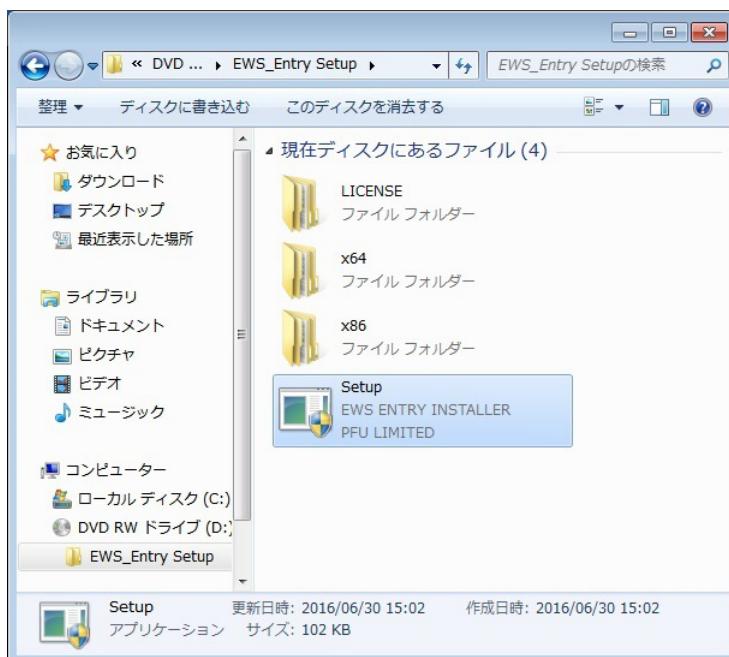
- Administrator 権限でログオンしている
- 実行中のアプリケーションがすべて終了している

◇インストールする

EmbedWare/SysMon Entry のインストール方法を以下に示します。

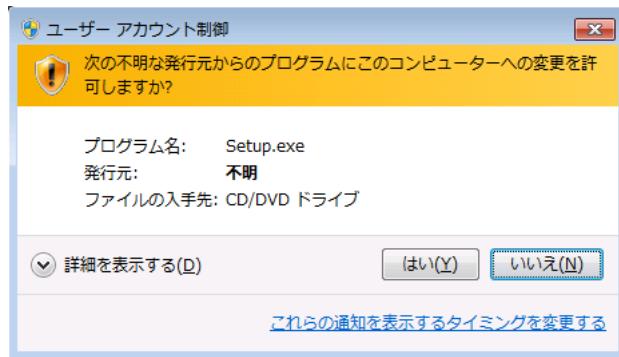
- 1 ダウンロードした「**EmbedWare/SysMon Entry Ver 7.1**」アーカイブファイルを、任意の作業用フォルダーに展開します。または、装置に添付されているドライバ CD をセットします。
- 2 手順 1 で展開した作業用フォルダー、または CD-ROM ドライブを選択し、「**EWS_Entry Setup**」フォルダーをダブルクリックします。

フォルダーの内容が表示されます。



3 "Setup.exe" をダブルクリックします。

以下のメッセージが表示された場合、[はい] ボタンをクリックします。



EmbedWare/SysMon Entry のインストーラーが起動されます。



インストーラー画面の [キャンセル] ボタンをクリックすると、インストールを中断できます。

a [キャンセル] ボタンをクリックします。

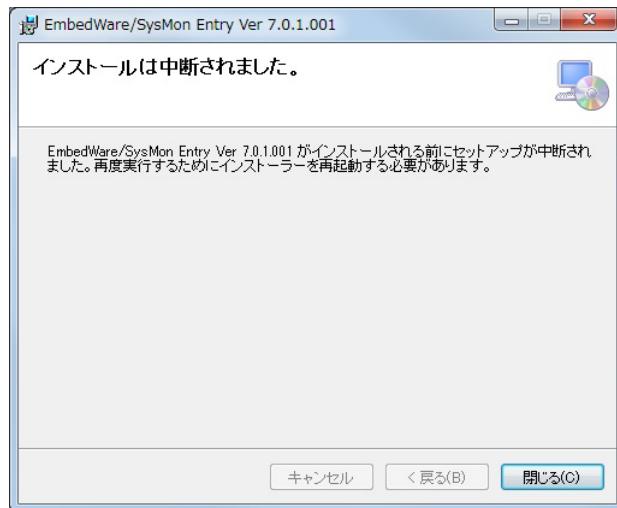
インストールの中止を確認するメッセージボックスが表示されます。





b [はい] ボタンをクリックします。

インストールが中断されたことを示す画面が表示されます。

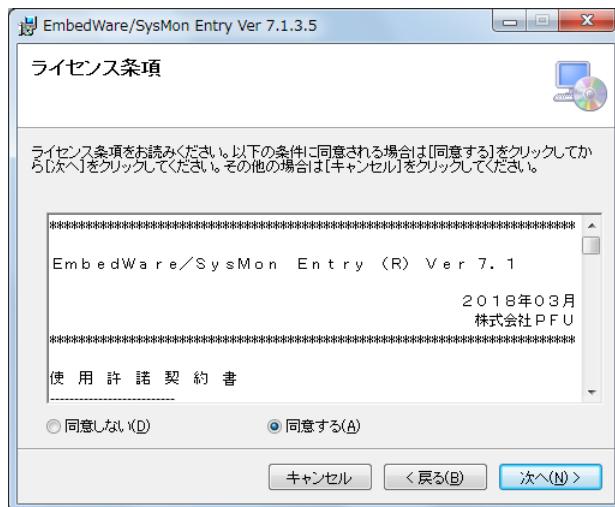


手順 a で [いいえ] ボタンをクリックすると、確認メッセージボックスが閉じられ、インストーラー画面に戻ります。

c [閉じる] ボタンをクリックします。

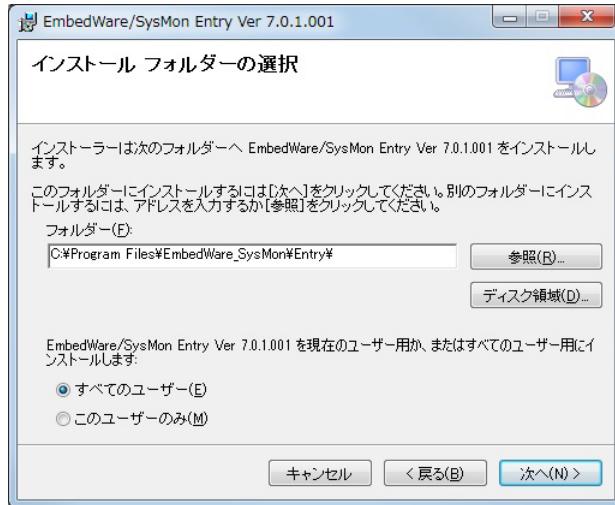
4 [次へ] ボタンをクリックします。

ライセンス条項画面が表示されます。



5 [同意する] ラジオボタンを選択し、[次へ] ボタンをクリックします。

インストールフォルダーの選択画面が表示されます。



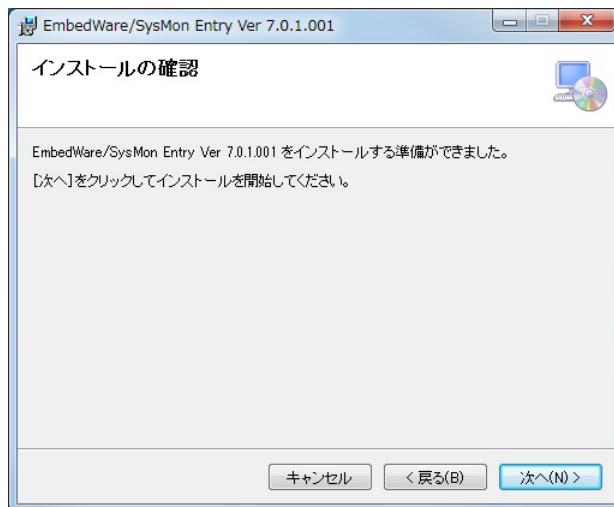
インストール先を変更する場合は、[参照...] ボタンをクリックして変更します。
ディスク情報を確認する場合は、[ディスク領域...] ボタンをクリックして、ディスク情報画面を表示します。



- ・ボリューム : 存在するディスクの名前が表示されます。
- ・ディスク サイズ : ディスクのサイズが表示されます。
- ・空き領域 : ディスクの空き領域が表示されます。
- ・必要な領域 : ディスクの使用領域が表示されます。
- ・残領域 : ディスクの残領域が表示されます。

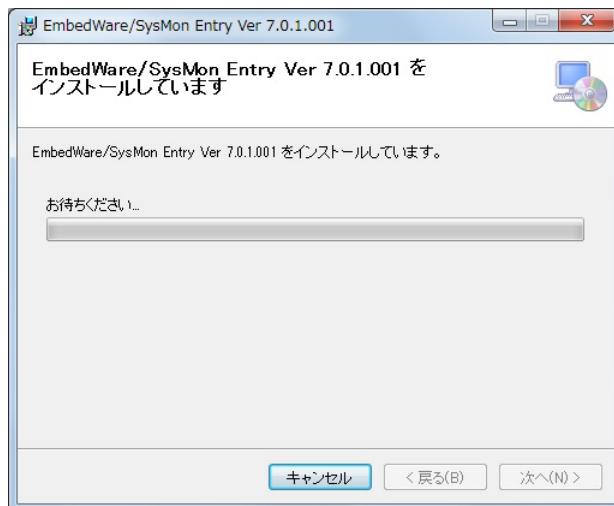
6 [次へ] ボタンをクリックします。

インストールの開始確認画面が表示されます。

**7 [次へ] ボタンをクリックします。**

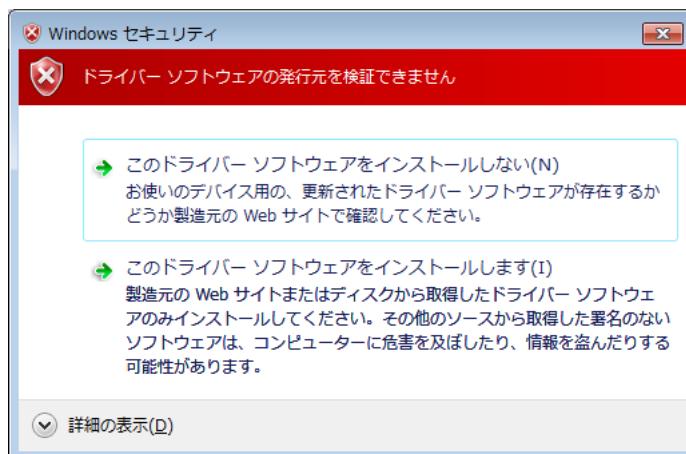
インストールが開始され、インストールの経過画面が表示されます。

インストールを中止する場合は、[キャンセル] ボタンをクリックします。

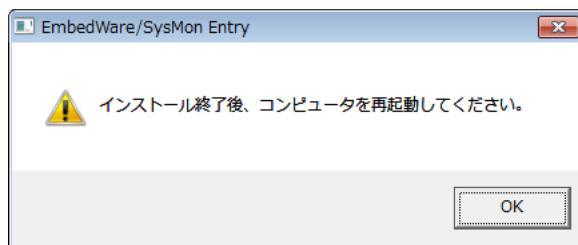




インストールの途中で、以下の画面が表示された場合は、必ず「このドライバーソフトウェアをインストールします」を選択して、インストールを継続してください。

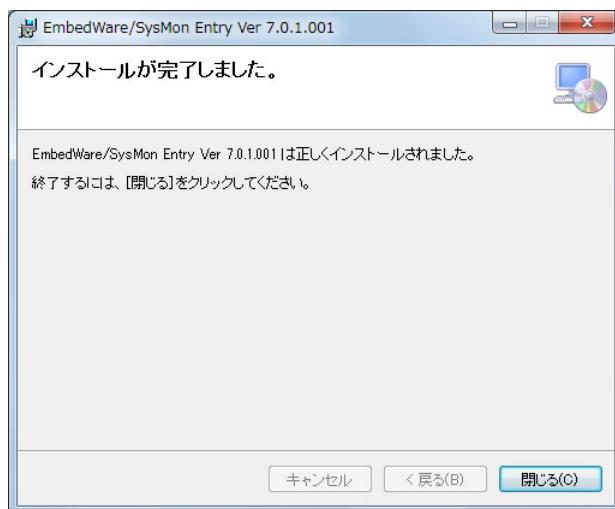


インストールが終わると、以下のメッセージが表示されます。



8 [OK] ボタンをクリックします。

インストールの完了画面が表示されます。



9 [閉じる] ボタンをクリックします。

10 コンピューターを再起動します。

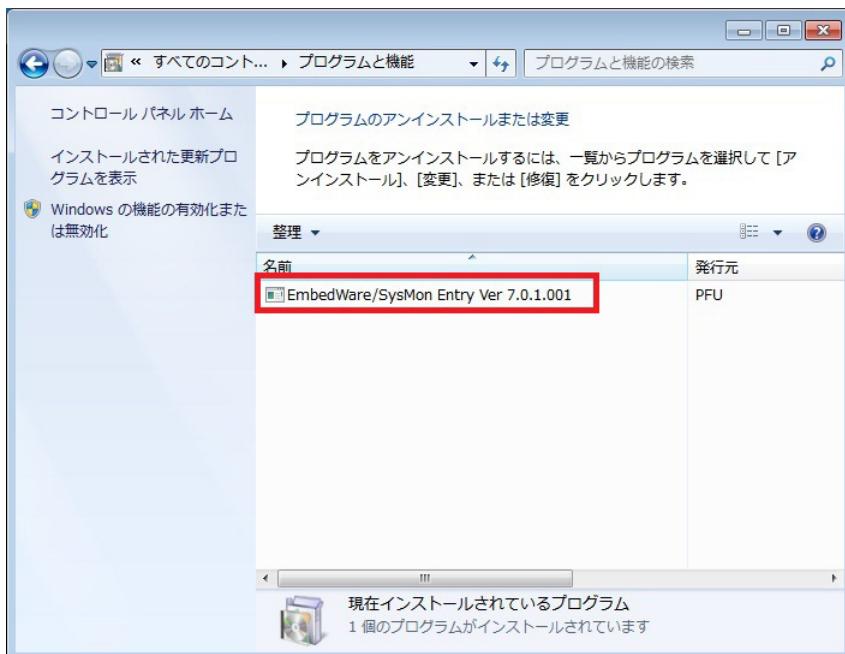
2.2 アンインストール

1 コントロールパネルから、EmbedWare/SysMon Entry の削除を起動します。

1 「スタート」 → 「コントロールパネル」をクリックします。

2 「プログラムのアンインストール」をクリックします。

3 「EmbedWare/SysMon Entry」をダブルクリックします。

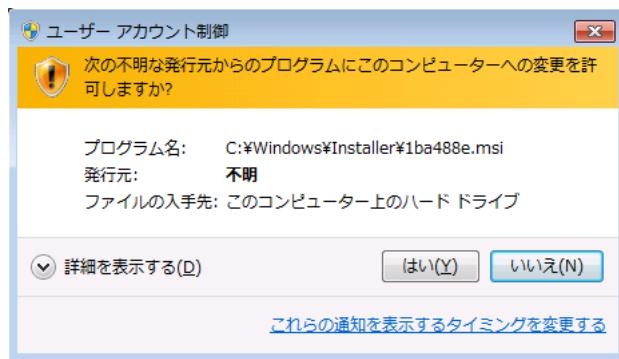


以下の確認メッセージが表示されます。

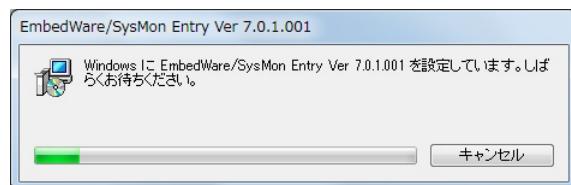


2 [はい] ボタンをクリックします。

以下のメッセージが表示された場合、[はい] ボタンをクリックします。



アンインストールが開始され、アンインストールの経過画面が表示されます。



アンインストールを中止すると不具合が生じる場合があるため、[キャンセル] ボタンをクリックしないでください。

[キャンセル] ボタンをクリックして不具合が生じた場合は、「スタート」→「コントロールパネル」→「プログラムのアンインストール」から、「EmbedWare/SysMon Entry」を選択し、[変更] ボタンをクリックして、プログラムの修復を行ってください。

それでも問題がある場合は、担当営業にご相談ください。

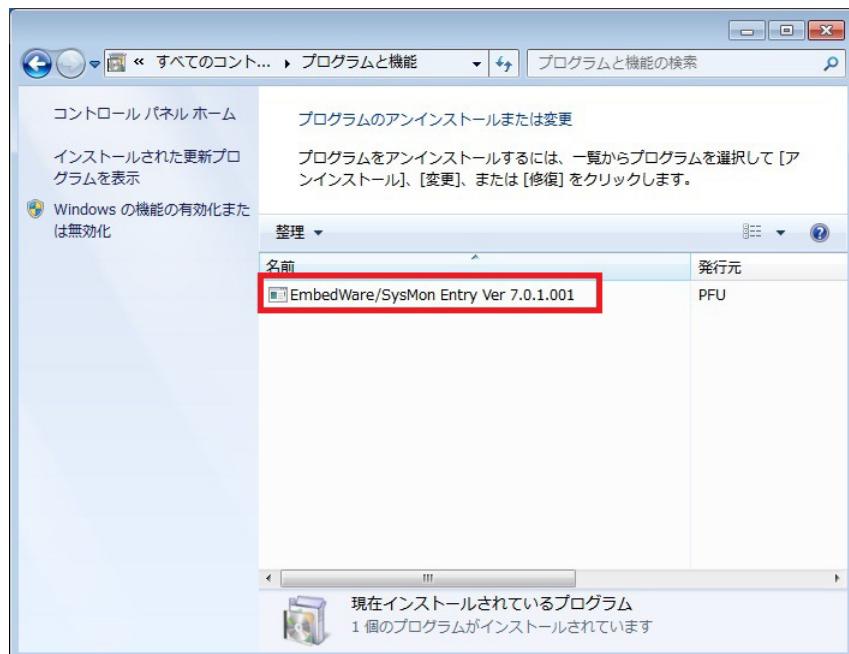
2.3 版数確認

以下の手順で版数を確認します。

1 「スタート」→「コントロールパネル」をクリックします。

2 「プログラムのアンインストール」をクリックします。

版数が表示されます。



第3章 EmbeddedWare/SysMon Entry を利用する

この章では、EmbeddedWare/SysMon Entry の利用方法について説明します。

デスクトップ上の「EmbeddedWare_SysMon Entry」のアイコンをダブルクリックすると、EmbeddedWare/SysMon Entry の画面が表示されます。

ここでは、EmbeddedWare/SysMon Entry のメニュー項目について説明します。

メニュー項目	説明
Temperature	温度監視
Voltage	電圧監視
Fan Speed	ファン監視
S.M.A.R.T.	内蔵ディスクデバイスの S.M.A.R.T. エラー監視
RAID	チップセット RAID および RAID カード（タイプ 1）監視
SVRx RAID	RAID カード（タイプ 2）監視
Software RAID	ソフトウェア RAID 監視
System Error	PCI、メモリ、BMC エラー監視
Information	システム情報表示
Performance	パフォーマンス情報表示
EmbeddedWare/SysMon Log	EmbeddedWare/SysMon イベントログの表示
BIOS Event Log	BIOS イベントログの表示
BMC SEL	BMC SEL の表示
WatchDog Timer	ウォッチ ドッグ タイマ設定
Options	Windows イベントログ出力、異常検出時の通知方法、EmbeddedWare/SysMon イベントログ出力 / ログ保存世代数、BIOS イベントログログ保存世代数、トレース機能、BMC SEL に関する設定



- 注意**
- 機種によって、一部の機能が存在しない場合があります。
 - Administrator 権限を持たないユーザーは、情報の参照だけ可能です。設定の変更および BMC SEL へのアクセスはできません。
 - EmbedWare/SysMon Entry を Administrator 権限で実行する場合は、起動時に「管理者として実行」を選択する必要があります。
 - BMC 搭載機種の温度監視、電圧監視、ファン監視で異常が発生したあとに、現在値が正常範囲に戻ったにもかかわらず、状態判定が異常のままになることがあります。状態判定に使用されるしきい値がヒステリシスを持っているためです。

3.1 溫度監視

(1) 溫度の表示

「Temperature」で、現在の温度、状態、温度のしきい値、過去最高値・過去最低値を確認できます。

[Item] にセンサー名が表示されます。

Item	Current Value	Lower Threshold	Upper Threshold	Lower Worst	Upper Worst
CPU_CORE0_temp	32°C		95°C	23°C 16/10/13 16:11:24	43°C 16/10/13 13:1...
CPU_CORE1_temp	32°C		95°C	25°C 16/10/13 14:05:07	46°C 16/10/13 13:1...
CPU_CORE2_temp	32°C		95°C	22°C 16/10/13 14:54:00	40°C 16/10/12 17:2...
CPU_CORE3_temp	31°C		95°C	23°C 16/10/13 16:01:56	41°C 16/10/12 17:2...
Air1_temp	29°C	0°C	51°C	23°C 16/10/13 16:01:56	34°C 16/08/18 16:1...
Air2_temp	28°C	0°C	51°C	23°C 16/10/13 16:01:56	33°C 16/08/18 16:1...

図 1 溫度の表示

- 現在の状態

[Item] に表示されているセンサー名の左側のアイコンが、現在の状態を表します。

● : 正常

● : 異常

○ : 監視対象外

- 現在の温度

[Current Value] に、現在の温度を表示します。

- 温度のしきい値（下限）

[Lower Threshold] に、温度のしきい値（下限）を表示します。



機種によって、温度のしきい値（下限）は存在しない場合があります。

- 温度のしきい値（上限）

[Upper Threshold] に、温度のしきい値（上限）を表示します。

- 過去最低値

[Lower Worst] に、温度の最低値の記録と発生日時を表示します。

- 過去最高値

[Upper Worst] に、温度の最高値の記録と発生日時を表示します。

各ボタンについて、以下に説明します。

[Monitoring] ボタン : 監視設定画面を表示します。詳細は、「(2) 監視条件の設定」(20 ページ) を参照してください

[Threshold Settings] ボタン : しきい値設定画面を表示します。詳細は、「(3) しきい値の設定」(21 ページ) を参照してください。

[Worst Reset] ボタン : 過去最高値・過去最低値を初期化します。

(2) 監視条件の設定

「Temperature」で〔Monitoring〕ボタンをクリックすると、監視設定画面が表示されます。

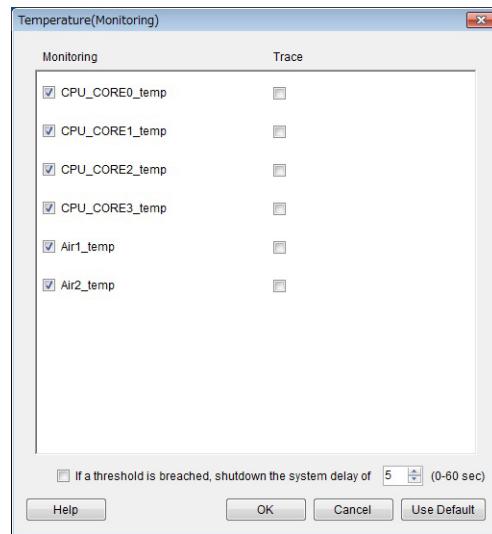


図2 監視条件の設定

監視設定画面では、以下の設定が可能です。

- ・ 監視対象項目を設定する

監視対象センサーの〔Monitoring〕チェックボックスをクリックして、チェックを付けます。監視対象外とする場合は、チェックを外します。

- ・ トレースを設定する

トレースを採取するセンサーの〔Trace〕チェックボックスをクリックして、チェックを付けます。採取しない場合は、チェックを外します。トレース情報（センサー名、現在の温度、記録した日付時刻）は、1秒間隔で「temperature.csv」ファイルに出力されます。ファイルはインストールフォルダーに格納されます。



トレースを採取するためには、事前にトレースの設定を行う必要があります。「3.15 オプション設定」(46 ページ) を参照し、Trace Function の設定を〔ON〕にしてください。

- ・ シャットダウンを設定する

しきい値を超える、または下回る異常を検出した場合に、コンピューターの電源を切断するかどうかを設定します。

切断する場合は、〔If a threshold is breached, shutdown the system delay of〕 チェックボックスをクリックして、チェックを付けます。切断しない場合は、チェックを外します。電源が切断されるまでの時間は、0 ~ 60 秒の間で設定します。



電源の切断は、ハードウェアの保護のため強制的に行われます。未保存の作業中ファイルの内容が失われる場合がありますので、ご注意ください。

各ボタンについて、以下に説明します。

- 〔Help〕ボタン : ヘルプ画面を表示します。
- 〔OK〕ボタン : 設定変更を保存し、監視設定画面を閉じます。
- 〔Cancel〕ボタン : 設定変更を保存せず、監視設定画面を閉じます。
- 〔Use Default〕ボタン : 電源が切断されるまでの時間を初期設定に戻します。

(3) しきい値の設定

「Temperature」で「Threshold Settings」ボタンをクリックすると、しきい値設定画面が表示されます。

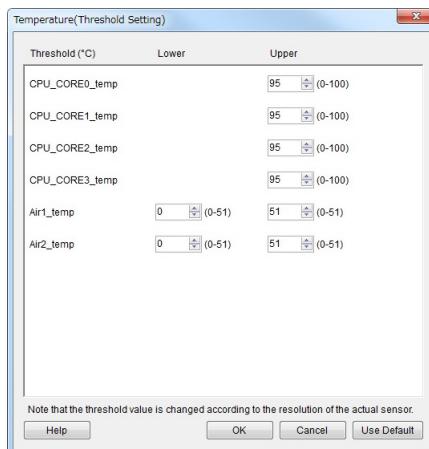


図 3 しきい値の設定

しきい値設定画面では、以下の設定が可能です。

- しきい値を設定する

温度を正常と判断する下限／上限値を、括弧内で示している範囲で変更できます。



BMC 搭載機種の場合、しきい値はセンサーの分解能に合わせて調整される場合があります。

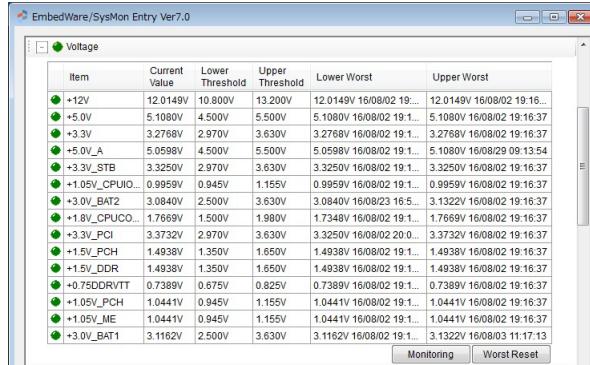
各ボタンについて、以下に説明します。

- 〔Help〕ボタン : ヘルプ画面を表示します。
- 〔OK〕ボタン : 設定変更を保存し、しきい値設定画面を閉じます。
- 〔Cancel〕ボタン : 設定変更を保存せず、しきい値設定画面を閉じます。
- 〔Use Default〕ボタン : しきい値を初期設定に戻します。

3.2 電圧監視

(1) 電圧の表示

「Voltage」で、現在の電圧、状態、電圧のしきい値、過去最高値・過去最低値を確認できます。
[Item] にセンサーネームが表示されます。



Item	Current Value	Lower Threshold	Upper Threshold	Lower Worst	Upper Worst
+12V	12.0149V	10.800V	13.200V	12.0149V 16/08/02 19:16...	12.0149V 16/08/02 19:16...
+5.0V	5.1080V	4.500V	5.500V	5.1080V 16/08/02 19:1...	5.1080V 16/08/02 19:16:37
+3.3V	3.2768V	2.970V	3.630V	3.2768V 16/08/02 19:1...	3.2768V 16/08/02 19:16:37
+5.0V_A	5.0598V	4.500V	5.500V	5.0598V 16/08/02 19:1...	5.1080V 16/08/29 09:13:54
+3.3V_STB	3.3250V	2.970V	3.630V	3.3250V 16/08/02 19:1...	3.3250V 16/08/02 19:16:37
+1.05V_CPU0...	0.9959V	0.945V	1.155V	0.9959V 16/08/02 19:1...	0.9959V 16/08/02 19:16:37
+3.0V_BAT2	3.0840V	2.500V	3.630V	3.0840V 16/08/23 16:5...	3.1322V 16/08/02 19:16:37
+1.05V_CPU0C...	1.7669V	1.500V	1.980V	1.7348V 16/08/02 19:1...	1.7669V 16/08/02 19:16:37
+3.3V_PCI	3.3732V	2.970V	3.630V	3.3250V 16/08/02 20:0...	3.3732V 16/08/02 19:16:37
+1.05V_PCH	1.4938V	1.350V	1.650V	1.4938V 16/08/02 19:1...	1.4938V 16/08/02 19:16:37
+1.05V_DDR	1.4938V	1.350V	1.650V	1.4938V 16/08/02 19:1...	1.4938V 16/08/02 19:16:37
+0.75DDRVTT	0.7389V	0.675V	0.825V	0.7389V 16/08/02 19:1...	0.7389V 16/08/02 19:16:37
+1.05V_LPC	1.0441V	0.945V	1.155V	1.0441V 16/08/02 19:1...	1.0441V 16/08/02 19:16:37
+1.05V_ME	1.0441V	0.945V	1.155V	1.0441V 16/08/02 19:1...	1.0441V 16/08/02 19:16:37
+3.0V_BAT1	3.1162V	2.500V	3.630V	3.1162V 16/08/02 19:1...	3.1322V 16/08/03 11:17:13

図 4 電圧の表示

・ 現在の状態

[Item] に表示されているセンサーネームの左側のアイコンが、現在の状態を表します。

● 正常

● 異常

○ 監視対象外

・ 現在の電圧

[Current Value] に、現在の電圧を表示します。

・ 電圧のしきい値（下限）

[Lower Threshold] に、電圧のしきい値（下限）を表示します。

・ 電圧のしきい値（上限）

[Upper Threshold] に、電圧のしきい値（上限）を表示します。

・ 過去最低値

[Lower Worst] に、電圧の最低値の記録と発生日時を表示します。

・ 過去最高値

[Upper Worst] に、電圧の最高値の記録と発生日時を表示します。

各ボタンについて、以下に説明します。

[Monitoring] ボタン : 監視設定画面を表示します。詳細は「(2) 監視条件の設定」(23 ページ) を参照してください。

[Worst Reset] ボタン : 過去最高値・過去最低値を初期化します。

(2) 監視条件の設定

「Voltage」で「Monitoring」ボタンをクリックすると、監視設定画面が表示されます。

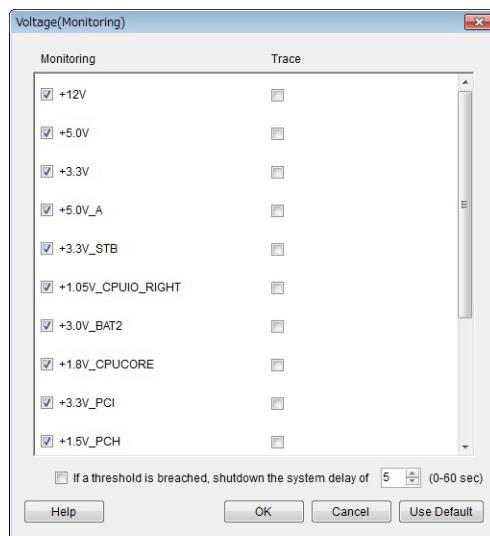


図 5 監視条件の設定

監視設定画面では、以下の設定が可能です。

- **監視対象項目を設定する**

監視対象センサーの「Monitoring」チェックボックスをクリックして、チェックを付けます。監視対象外とする場合は、チェックを外します。

- **トレースを設定する**

トレースを採取するセンサーの「Trace」チェックボックスをクリックして、チェックを付けます。採取しない場合は、チェックを外します。トレース情報（センサー名、現在の電圧、記録した日付時刻）は、1秒間隔で「voltage.csv」ファイルに出力されます。ファイルはインストールフォルダーに格納されます。



トレースを採取するためには、事前にトレースの設定を行う必要があります。「3.15 オプション設定」(46 ページ) を参照し、Trace Function の設定を「ON」にしてください。

- **シャットダウンを設定する**

しきい値を超える、または下回る異常を検出した場合に、コンピューターの電源を切断するかどうかを設定します。

切断する場合は、「If a threshold is breached, shutdown the system delay of」チェックボックスをクリックして、チェックを付けます。切断しない場合は、チェックを外します。

電源が切断されるまでの時間は、0 ~ 60 秒の間で設定します。



電源の切断は、ハードウェアの保護のため強制的に行われます。未保存の作業中ファイルの内容が失われる場合がありますので、ご注意ください。

各ボタンについて、以下に説明します。

- [Help] ボタン : ヘルプ画面を表示します。
- [OK] ボタン : 設定変更を保存し、監視設定画面を閉じます。
- [Cancel] ボタン : 設定変更を保存せず、監視設定画面を閉じます。
- [Use Default] ボタン : 電源が切断されるまでの時間を初期設定に戻します。

3.3 ファン監視

(1) ファン回転数の表示

「Fan Speed」で、現在のファン回転数、状態、ファン回転数のしきい値、過去最高値・過去最低値を確認できます。[Item] にセンサー名が表示されます。

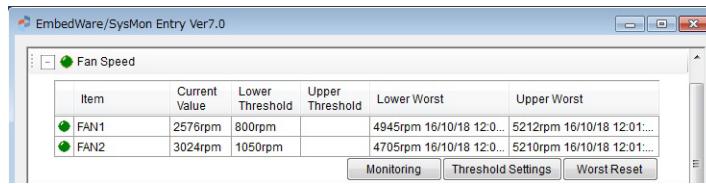


図 6 ファン回転数の表示

- 現在の状態

[Item] に表示されているセンサー名の左側のアイコンが、現在の状態を表します。

● 正常

● 異常

○ 監視対象外

- 現在のファン回転数

[Current Value] に、現在のファン回転数を表示します。

- ファン回転数のしきい値（下限）

[Lower Threshold] に、ファン回転数のしきい値（下限）を表示します。

- ファン回転数のしきい値（上限）

[Upper Threshold] に、ファン回転数のしきい値（上限）を表示します。



機種によって、ファン回転数のしきい値（上限）は存在しない場合があります。

備考

- 過去最低値

[Lower Worst] に、ファン回転数の最低値の記録と発生日時を表示します。

- 過去最高値

[Upper Worst] に、ファン回転数の最高値の記録と発生日時を表示します。

各ボタンについて、以下に説明します。

[Monitoring] ボタン : 監視設定画面を表示します。詳細は、「(2) 監視条件の設定」(26 ページ) を参照してください。

[Threshold Settings] ボタン : しきい値設定画面を表示します。詳細は、「(3) しきい値の設定」(27 ページ) を参照してください。

[Worst Reset] ボタン : 過去最高値・過去最低値を初期化します。

(2) 監視条件の設定

「Fan Speed」で「Monitoring」ボタンをクリックすると、監視設定画面が表示されます。

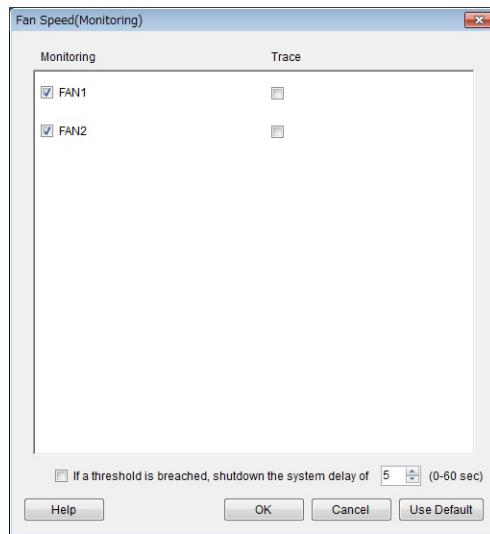


図7 監視条件の設定

監視設定画面では、以下の設定が可能です。

- **監視対象項目を設定する**

監視対象センサーの「Monitoring」チェックボックスをクリックして、チェックを付けます。監視対象外とする場合は、チェックを外します。

- **トレースを設定する**

トレースを採取するセンサーの「Trace」チェックボックスをクリックして、チェックを付けます。採取しない場合は、チェックを外します。トレース情報（センサー名、現在のファン回転数、記録した日付時刻）は、1秒間隔で「fan.csv」ファイルに出力されます。ファイルはインストールフォルダーに格納されます。



トレースを採取するためには、事前にトレースの設定を行う必要があります。「3.15 オプション設定」(46 ページ) を参照し、Trace Function の設定を「ON」にしてください。

- **シャットダウンを設定する**

しきい値を超える、または下回る異常を検出した場合に、コンピューターの電源を切断するかどうかを設定します。

切斷する場合は、「If a threshold is breached, shutdown the system delay of」チェックボックスをクリックして、チェックを付けます。切斷しない場合は、チェックを外します。

電源が切斷されるまでの時間は、0 ~ 60 秒の間で設定します。



電源の切斷は、ハードウェアの保護のため強制的に行われます。未保存の作業中ファイルの内容が失われる場合がありますので、ご注意ください。

各ボタンについて、以下に説明します。

〔Help〕ボタン : ヘルプ画面を表示します。

〔OK〕ボタン : 設定変更を保存し、監視設定画面を閉じます。

〔Cancel〕ボタン : 設定変更を保存せず、監視設定画面を閉じます。

〔Use Default〕ボタン : 電源が切斷されるまでの時間を初期設定に戻します。

(3) しきい値の設定

「Fan Speed」で「Threshold Settings」ボタンをクリックすると、しきい値設定画面が表示されます。

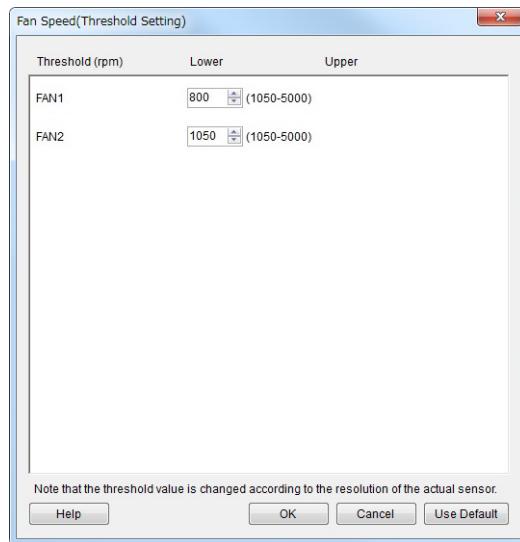


図 8 しきい値の設定

しきい値設定画面では、以下の設定が可能です。

- しきい値を設定する

ファン回転数を正常と判断する下限／上限値を、括弧内で示している範囲で変更できます。



BMC 搭載機種の場合、しきい値はセンサーの分解能に合わせて調整される場合があります。

各ボタンについて、以下に説明します。

- | | |
|-------------------|----------------------------|
| [Help] ボタン | : ヘルプ画面を表示します。 |
| [OK] ボタン | : 設定変更を保存し、しきい値設定画面を閉じます。 |
| [Cancel] ボタン | : 設定変更を保存せず、しきい値設定画面を閉じます。 |
| [Use Default] ボタン | : しきい値を初期設定に戻します。 |

3.4 S.M.A.R.T. 監視

(1) S.M.A.R.T. ステータスの表示

「S.M.A.R.T.」で、ディスクの S.M.A.R.T. ステータスを確認できます。
S.M.A.R.T. をサポートしている物理ディスクがセンサー名 SLOT0 ~ SLOT7 で表示されます。



装置が AR シリーズの場合、SLOT の番号は装置に印字されているハードディスクの番号に対応します。

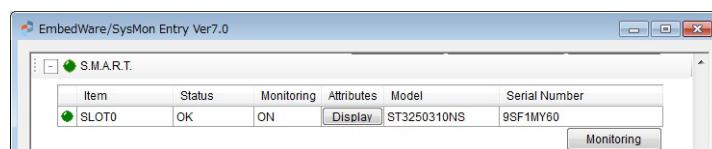


図9 S.M.A.R.T. ステータスの表示

- 現在の状態

- [Item] に表示されているセンサー名の左側のアイコンが、現在の状態を表します。
- : OK (正常)
 - : WARNING (S.M.A.R.T. 機能により異常が通知されています)
 - : UNKNOWN (不明、または S.M.A.R.T. 機能がサポートされていません)、監視対象外



チップセット内蔵 RAID コントローラーが有効になっている場合、状態は UNKNOWN になります。S.M.A.R.T. アトリビュート情報も取得できません。

- 監視状態

[Monitoring] に現在監視が有効になっているかどうかを表示します。

- | | |
|-----|---------|
| ON | : 監視対象 |
| OFF | : 監視対象外 |

- ディスクのモデル

[Model] にディスクのモデル名を表示します。

- ディスクのシリアル番号

[Serial Number] にディスクのシリアル番号を表示します。

各ボタンについて、以下に説明します。

[Monitoring] ボタン : 監視設定画面を表示します。詳細は、「(2) 監視設定」(29 ページ) を参照してください。

[Display] ボタン : 各ディスクの S.M.A.R.T. アトリビュート情報を表示します。詳細は、「(3) S.M.A.R.T. アトリビュート情報の表示」(30 ページ) を参照してください。

(2) 監視設定

「S.M.A.R.T.」で〔Monitoring〕ボタンをクリックすると、監視設定画面が表示されます。

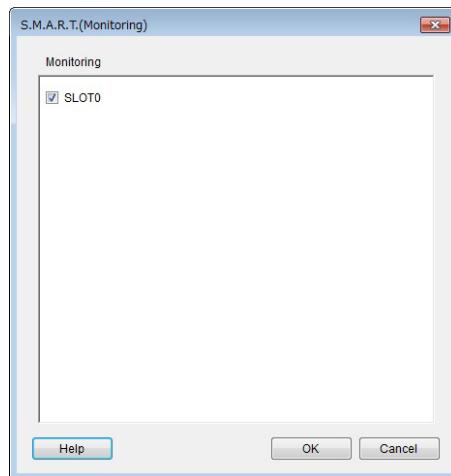


図 10 監視設定

監視設定画面では、以下の設定が可能です。

- **監視対象項目を設定する**

監視対象ディスクの〔Monitoring〕チェックボックスをクリックして、チェックを付けます。監視対象外とする場合は、チェックを外します。

各ボタンについて、以下に説明します。

- | | |
|--------------|--------------------------|
| 〔Help〕 ボタン | ： ヘルプ画面を表示します。 |
| 〔OK〕 ボタン | ： 設定変更を保存し、監視設定画面を閉じます。 |
| 〔Cancel〕 ボタン | ： 設定変更を保存せず、監視設定画面を閉じます。 |

(3) S.M.A.R.T. アトリビュート情報の表示

「S.M.A.R.T.」で「Display」ボタンをクリックすると、各ディスクのS.M.A.R.T.アトリビュート情報が表示されます。

図 11 S.M.A.R.T. アトリビュート情報の表示

各ボタンについて、以下に説明します。

〔Save As...〕 ボタン 表示内容をファイルに保存します。
〔OK〕 ボタン 画面を閉じます。

3.5 RAID 監視

(1) RAID 情報の表示

「RAID」で、チップセット RAID および RAID カード（タイプ 1）の状態を確認できます。

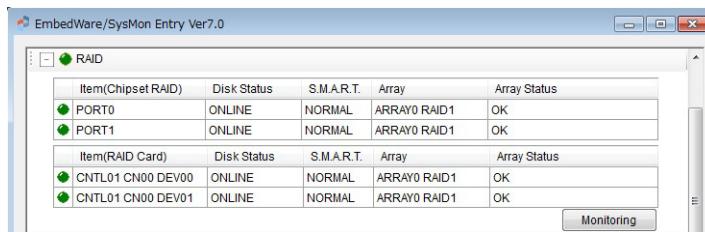


図 12 RAID 情報の表示

• RAID 監視種別

以下の RAID 種別が表示されます。

Item (Chipset RAID) : 監視対象はチップセット内蔵 RAID コントローラーです。

Item (RAID Card) : 監視対象は RAID カード（タイプ 1）です。

• 物理ディスク状態

デバイス名の左側のアイコンが、現在の物理ディスクの状態を表します。

● : ONLINE/HOTSPARE (正常)

○ : REBUILDING (リビルド中)

● : FAILED (ディスク故障、または S.M.A.R.T. による異常検出)

○ : NOARRAY (RAID 構築されていません)

• デバイス名

チップセット内蔵 RAID コントローラの場合、PORT（ポート番号）で表示します。

RAID カード（タイプ 1）の場合、CNTL（コントローラー番号）、CN（コネクター番号）、DEV（デバイス番号）で表示します。

• S.M.A.R.T. ステータス

〔S.M.A.R.T.〕に物理ディスクの S.M.A.R.T. ステータスを表示します。

NORMAL : 正常

ERROR : 異常

N/A : S.M.A.R.T. 機能がサポートされていません

• 論理ボリューム情報

〔Array〕に各物理ディスクの所属する論理ボリュームの番号と RAID レベルを表示します。

• 論理ボリューム状態

〔Array Status〕に各物理ディスクの所属する論理ボリュームの状態を表示します。

OK (背景色:なし) : 論理ボリュームの状態が正常

N/A (背景色:なし) : 不明

REBUILDING/DEGRADED (背景色:黄) : 論理ボリュームの冗長性が失われており、リビルドが完了していません。

FAILED (背景色:赤) : 論理ボリュームが修復不能です。

各ボタンについて、以下に説明します。

〔Monitoring〕ボタン：監視設定画面を表示します。詳細は「(2) 監視設定」(32 ページ) を参照してください。

(2) 監視設定

「RAID」で〔Monitoring〕ボタンをクリックすると、監視設定画面が表示されます。

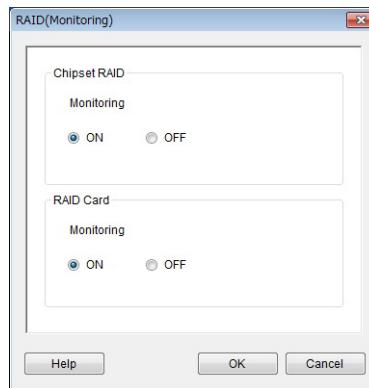


図 13 監視設定

監視設定画面では、以下の設定が可能です。

- **監視の有効・無効を設定する**

監視を有効とする場合は、[ON] のラジオボタンを選択します。

監視を無効とする場合は、[OFF] のラジオボタンを選択します。

各ボタンについて、以下に説明します。

〔Help〕ボタン：ヘルプ画面を表示します。

〔OK〕ボタン：設定変更を保存し、監視設定画面を閉じます。

〔Cancel〕ボタン：設定変更を保存せず、監視設定画面を閉じます。



RAID 監視の有効・無効の設定変更は、システム再起動後に有効となります。

3.6 SVRx RAID 監視

(1) SVRx RAID 情報の表示

「SVRx RAID」で、RAID カード（タイプ2）の情報を確認できます。

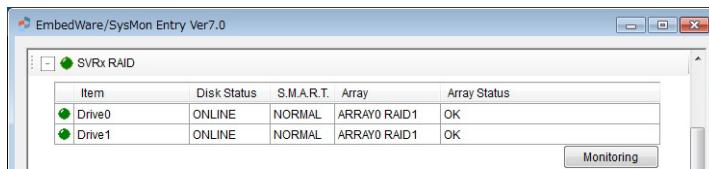


図 14 SVRx RAID 情報の表示

- 物理ディスク状態
デバイス名の左側のアイコンが、現在の物理ディスクの状態を表します。
● : ONLINE (正常)
○ : REBUILDING (リビルド中)
● : FAILED (ディスク故障、または S.M.A.R.T. による異常検出)
- デバイス名
Drive (ディスク番号) で表示します。
- S.M.A.R.T. ステータス
[S.M.A.R.T.] に物理ディスクの S.M.A.R.T. ステータスを表示します。
NORMAL : 正常
ERROR : 異常
N/A : S.M.A.R.T. 機能がサポートされていません
- 論理ボリューム情報
[Array] に各物理ディスクの所属する論理ボリュームの番号と RAID レベルを表示します。



論理ボリュームの番号は、OS の認識順です。

備考

- 論理ボリューム状態
[Array Status] に各物理ディスクの所属する論理ボリュームの状態を表示します。
OK (背景色: なし) : 論理ボリュームの状態が正常
N/A (背景色: なし) : 不明
REBUILDING/DEGRADED (背景色: 黄) : 論理ボリュームの冗長性が失われており、リビルドが完了していません。
FAILED (背景色: 赤) : 論理ボリュームが修復不能です。

各ボタンについて、以下に説明します。

[Monitoring] ボタン : 監視設定画面を表示します。詳細は「(2) 監視設定」(34 ページ) を参照してください。

(2) 監視設定

「SVRx RAID」で〔Monitoring〕ボタンをクリックすると、監視設定画面が表示されます。



図 15 監視設定

監視設定画面では、以下の設定が可能です。

- **監視の有効・無効を設定する**

監視を有効とする場合は、[ON] のラジオボタンを選択します。

監視を無効とする場合は、[OFF] のラジオボタンを選択します。

各ボタンについて、以下に説明します。

[Help] ボタン : ヘルプ画面を表示します。

[OK] ボタン : 設定変更を保存し、監視設定画面を閉じます。

[Cancel] ボタン : 設定変更を保存せず、監視設定画面を閉じます。



SVRx RAID 監視の有効・無効の設定変更は、システム再起動後に有効となります。

3.7 ソフトウェア RAID 監視

(1) ソフトウェア RAID 情報の表示

「Software RAID」で、ソフトウェア RAID の情報を確認できます。

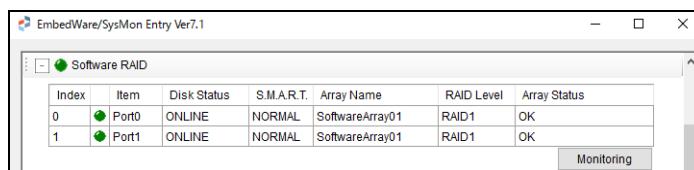


図 16 ソフトウェア RAID 情報の表示

- **Index**
ソフトウェア RAID を構成する物理ディスクの識別番号です。0 から始まる連続した番号です。
SDK を使用してソフトウェア RAID の情報を取得する際に使用します。
- **アイコン**
物理ディスクのステータスを表します。
 - : 正常
 - : リビルド中
 - : ディスク故障、または S.M.A.R.T. による異常検出
 - : RAID 構築なし
- **Item**
物理ディスクのポート番号です。
- **Disk Status**
物理ディスクの状態です。

ONLINE	: 正常
FAILED	: ディスク故障やディスク非搭載などの要因で、RAID から切り離されている
REBUILDING	: リビルド中
NOARRAY	: RAID の解体が予約されている（論理ボリュームのデータを引き継がないディスク）
PRE INIT	: RAID の解体が予約されている（論理ボリュームのデータを引き継ぐディスク）
- **S.M.A.R.T.**
物理ディスクの S.M.A.R.T. ステータスです。

NORMAL	: 正常
ERROR	: 異常
N/A	: S.M.A.R.T. ステータス取得不可
- **Array Name**
RAID の名前です。
- **RAID Level**
RAID レベルです。

- **Array Status**

RAID のステータスです。

OK (背景色: なし) : 正常

DEGRADED : 冗長性が失われている

REBUILDING <N>% : リビルド中、N は進捗率 (0 ~ 100)

FAILED : RAID 修復不能

PRE DELETE : RAID の解体が予約されている

各ボタンについて、以下に説明します。

〔Monitoring〕ボタン : 監視設定画面を表示します。詳細は「(2) 監視設定」(36 ページ) を参照してください。

(2) 監視設定

「Software RAID」で〔Monitoring〕ボタンをクリックすると、監視設定画面が表示されます。

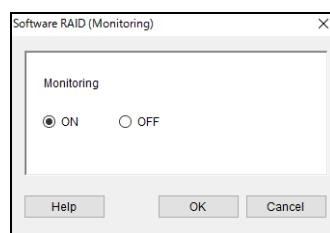


図 17 監視設定

監視設定画面では、以下の設定が可能です。

- **監視の有効・無効を設定する**

監視を有効とする場合は、〔ON〕のラジオボタンを選択します。

監視を無効とする場合は、〔OFF〕のラジオボタンを選択します。

各ボタンについて、以下に説明します。

〔Help〕ボタン : ヘルプ画面を表示します。

〔OK〕ボタン : 設定変更を保存し、監視設定画面を閉じます。

〔Cancel〕ボタン : 設定変更を保存せず、監視設定画面を閉じます。

3.8 システムエラー監視

(1) システムエラー情報の表示

「System Error」で、装置起動時に取得した BIOS イベントログから検出したエラーの情報を確認できます。

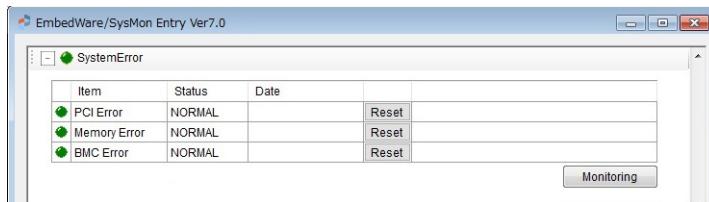


図 18 システムエラー情報の表示

• PCI Error

PCI Error に関するエラーログの有無と、エラーログがある場合、最初のログの日時を表示します。[Item] に表示されている PCI Error の左側のアイコンが、現在の状態を表します。

- : エラーなし (NORMAL)
- : エラーあり (ERROR)
- : 監視対象外

• Memory Error

Memory Error に関するエラーログの有無と、エラーログがある場合、最初のログの日時を表示します。[Item] に表示されている Memory Error の左側のアイコンが、現在の状態を表します。

- : エラーなし (NORMAL)
- : エラーあり (ERROR)
- : 監視対象外

• BMC Error

BMC Error に関するエラーログの有無と、エラーログがある場合、最初のログの日時を表示します。[Item] に表示されている BMC Error の左側のアイコンが、現在の状態を表します。

- : エラーなし (NORMAL)
- : エラーあり (ERROR)
- : 監視対象外

各ボタンについて、以下に説明します。

[Reset] ボタン : BIOS イベントログから検出したエラーの状態を正常に戻します。新規のエラーが検出された場合は、再びエラーが検出されるようになります。



- BIOS イベントログを消去しても検出済みの PCI、メモリ、BMC エラーは、エラー状態のリセットをしない限り、エラー検出状態を解除しません。詳細は「付録 D システムエラー監視の仕組みについて」(113 ページ) を参照してください。
- エラー検出状態である場合、同一のエラーが再度発生しても無視されます。

〔Monitoring〕ボタン：監視設定画面を表示します。詳細は「(2) 監視設定」(38 ページ) を参照してください。

(2) 監視設定

「System Error」で〔Monitoring〕ボタンをクリックすると、監視設定画面が表示されます。

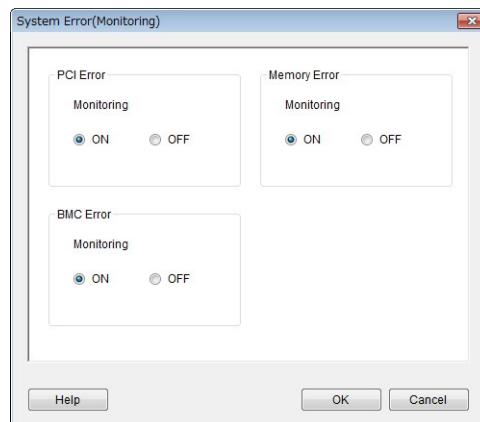


図 19 監視設定

監視設定画面では、以下の設定が可能です。

- **監視の有効・無効を設定する**

監視を有効とする場合は、〔ON〕のラジオボタンを選択します。

監視を無効とする場合は、〔OFF〕のラジオボタンを選択します。

各ボタンについて、以下に説明します。

〔Help〕ボタン : ヘルプ画面を表示します。

〔OK〕ボタン : 設定変更を保存し、監視設定画面を閉じます。

〔Cancel〕ボタン : 設定変更を保存せず、監視設定画面を閉じます。

3.9 システム情報表示

[Information] で、このシステムの各種情報を確認できます。



図 20 システム情報の表示

- BIOS Information

装置固有のカスタマイズコードと、BIOS 版数情報を表示します。

- Total uptime hours

装置の総通電時間を表示します。



この時間は、EmbedWare/SysMon Entry のサービス動作時間の累積となるため、運用環境によっては、正確に表示できない場合があります。

各ボタンについて、以下に説明します。

[Reset] ボタン : 総通電時間をリセットし、0.0 に戻します。

3.10 パフォーマンス情報

(1) パフォーマンス情報の表示

〔Performance〕で、CPU 実周波数を確認できます。

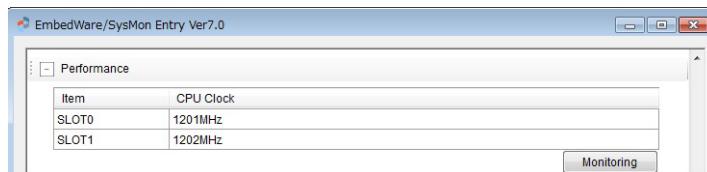


図 21 パフォーマンス情報の表示

- CPU パフォーマンス情報

物理 CPU ごとの実周波数 (MHz) を表示します。

各ボタンについて、以下に説明します。

〔Monitoring〕ボタン : 監視設定画面を表示します。詳細は「(2) 監視設定」(40 ページ) を参照してください。

(2) 監視設定

「Performance」で 〔Monitoring〕ボタンをクリックすると、監視設定画面が表示されます。

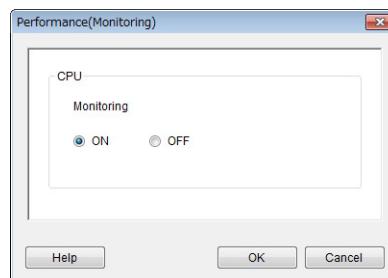


図 22 監視設定

監視設定画面では、以下の設定が可能です。

- 監視の有効・無効を設定する

監視を有効とする場合は、〔ON〕のラジオボタンを選択します。

監視を無効とする場合は、〔OFF〕のラジオボタンを選択します。

各ボタンについて、以下に説明します。

〔Help〕ボタン : ヘルプ画面を表示します。

〔OK〕ボタン : 設定変更を保存し、監視設定画面を閉じます。

〔Cancel〕ボタン : 設定変更を保存せず、監視設定画面を閉じます。

3.11 EmbedWare/SysMon イベントログ表示

「EmbedWare/SysMon Log」をクリックすると、EmbedWare/SysMon イベントログを確認できます。

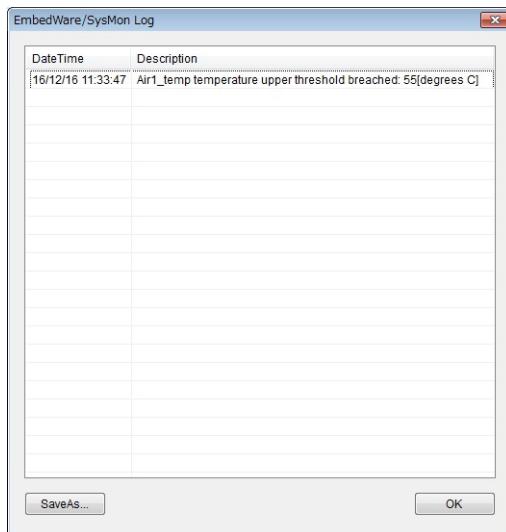


図 23 EmbedWare/SysMon イベントログの表示

- ログ世代管理について

EmbedWare/SysMon イベントログは、インストールフォルダー配下の「sysmon.log」に記録されます。ファイルサイズが 1MB に達すると、sysmon.log の末尾に数字を追加した名前に変名し、新たに sysmon.log が作成されます。ファイル名の数字が大きい方が、古い世代となります。

作成されるファイルの例 : sysmon.log.0

- 表示内容について

sysmon.log の内容が表示されます。

各ボタンについて、以下に説明します。

[Save As...] ボタン : 表示内容をファイルに保存します。

[OK] ボタン : 画面を閉じます。

3.12 BIOS イベントログ表示

「BIOS Event Log」をクリックすると、BIOS イベントログを確認できます。

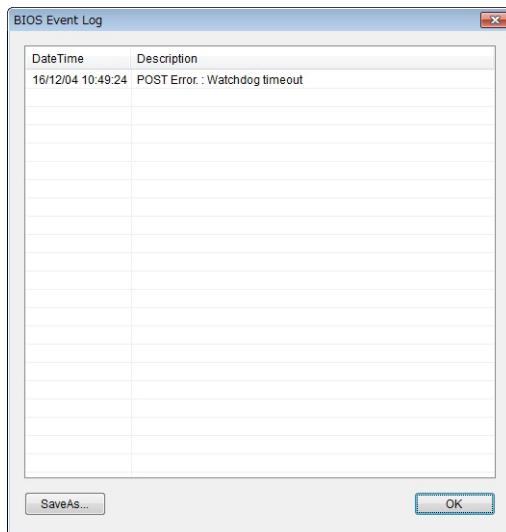


図 24 BIOS イベントログの表示

- ログ世代管理について

BIOS イベントログはインストールフォルダー配下の「biosevent.log」に記録されます。ファイルサイズが 1MB に達すると、biosevent.log の末尾に数字を追加した名前に変名し、新たに biosevent.log が作成されます。ファイル名の数字が大きい方が、古い世代となります。

作成されるファイルの例 : biosevent.log.0

- 表示内容について

biosevent.log の内容が表示されます。

各ボタンについて、以下に説明します。

[Save As...] ボタン : 表示内容をファイルに保存します。

[OK] ボタン : 画面を閉じます。

3.13 BMC SEL 表示

「BMC SEL」をクリックすると、BMC SEL の採取・表示・保存を行う画面が開きます。

図 25 BMC SEL の表示

- **BMC SEL の表示**
採取済みの BMC SEL を表示します。採取していない場合、BMC SEL は表示されません。
 - **採取時間の表示**
[Collection Time] に、表示している BMC SEL を採取した日時を表示します。

各ボタンについて、以下に説明します。

「Help」ボタン : ヘルプ画面を表示します。

「Save As...」ボタン : 表示内容をファイルに保存します。

「Start BMC SEL Collection】 ボタン : BMC SEL の採取を開始します。

本機能は Administrator 権限で有効です。採取済みのデータは、再び BMC SEL を採取することで上書きされます。

「OK」ボタン : 画面を閉じます

3.14 ウオッチドッグタイマ設定

「WatchDog Timer」をクリックすると、ウォッチドッグタイマ設定画面を表示します。

ウォッチドッグタイマとは、装置の正常な実行状態を監視するためのタイマです。

設定したタイムアウト時間内に EmbedWare/SysMon のサービスプログラムがウォッチドッグタイマにアクセスできなかった場合に、システムの状態が異常と判断され、ハードウェアによって指定したアクションが実施されます。初期値は [OFF] となります。

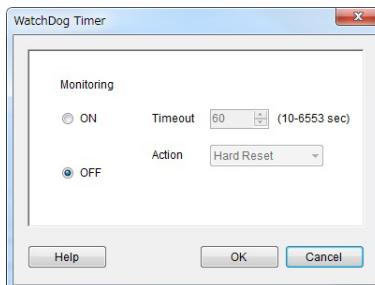


図 26 ウォッチドッグタイマの設定

ウォッチドッグタイマを設定する方法を、以下に示します。

1 [ON] ラジオボタンを選択し、タイムアウト時間およびタイムアウト発生時のアクションを設定します。

ウォッチドッグタイマを使用しない場合は、[OFF] を選択してください。

ウォッチドッグタイマのタイムアウト発生時のアクションについて、以下に説明します。

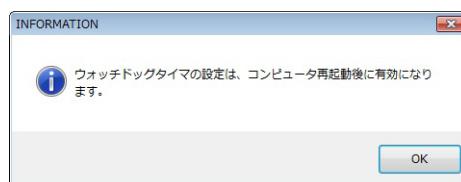
- Hard Reset : 装置再起動を実施します。
- No Action : 何も実施しません。
- Power Down : 装置のシャットダウンを実施します。
- Power Cycle : 装置のシャットダウンを実施したあとに装置起動を実施します。



- 装置でサポートしているアクションが表示されます。
- タイムアウト時間初期値は 60 秒、タイムアウト発生時のアクションの初期値は Hard Reset となります。

2 [OK] ボタンをクリックします。

設定内容に変更がある場合は、次のメッセージが表示されます。設定変更は、再起動後に有効になります。



CPU 使用率が継続して高負荷になるような操作 および OS 停止要因となる操作を実施する場合は、ウォッチドッグタイマの設定をオフにするか、タイムアウト値を適切な値に変更してください。

EmbedWare/SysMon サービスプログラムがウォッチドッグタイマにアクセスできず、装置がシャットダウンまたは再起動する可能性があります。

各ボタンについて、以下に説明します。

- 〔Help〕ボタン : ヘルプ画面を表示します。
- 〔OK〕ボタン : 設定変更を保存し、ウォッヂドッグタイマ設定画面を閉じます。
- 〔Cancel〕ボタン : 設定変更を保存せず、ウォッヂドッグタイマ設定画面を閉じます。

3.15 オプション設定

[Options] ボタンをクリックすると、各種設定を確認できます。

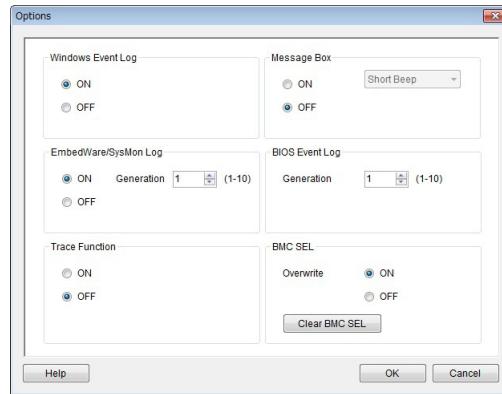


図 27 設定の表示

• Windows Event Log

監視項目の状態変化を検出した場合、Windows イベントログへ出力を行うかどうかを設定します。

ON : Windows イベントログへの出力を行います（初期値）。

OFF : Windows イベントログへの出力を行いません。

• Message Box

すべての監視項目について、異常検出時にデスクトップにメッセージボックスを表示するかどうかを設定します。

ON : メッセージボックスを表示します。

OFF : メッセージボックスを表示しません（初期値）。

上記設定が [ON] の場合、以下の項目を選択できます。

Short Beep : メッセージボックス表示時にビープ音を 3 秒間鳴らします。

Long Beep : メッセージボックス表示時にビープ音を鳴らし、[OK] ボタンをクリックするまで鳴らし続けます。

• EmbedWare/SysMon Log

EmbedWare/SysMon Log へ出力を行うかどうかを設定します。

ON : EmbedWare/SysMon Log への出力を行います（初期値）。

OFF : EmbedWare/SysMon Log への出力を行いません。

上記設定が [ON] の場合、以下の項目を設定できます。

Generation : ログ保存世代数を設定します（初期値：1 世代）。

- **BIOS Event Log**

BIOS イベントログの保存世代数を設定できます。

Generation : ログ保存世代数を設定します（初期値：1 世代）。



EmbedWare/SysMon イベントログおよび BIOS イベントログのログ保存世代数の初期値は 1 になります。

- **Trace Function**

温度、電圧、ファン監視機能について、測定値の履歴をトレースファイルに残すかどうかを選択します。

ON : トレースファイルを残します。

OFF : トレースファイルを残しません（初期値）。

- **BMC SEL**

BMC SEL 操作を行います。本機能は Administrator 権限で有効です。

Overwrite : BMC SEL の記録サイズが最大値に達した場合の動作を設定します。記録の上書きをする場合は、[ON] を選択してください。古い情報から上書きされます。記録の上書きをしない場合は、[OFF] を選択してください（初期値は [ON] です）。



記録の上書きをしない運用にした場合、記録サイズが最大値に達すると、BMC SEL に新たな記録を行うことができなくなります。

Clear BMC SEL : BMC 内の SEL 情報を削除します。



BMC 内の SEL 情報削除は、途中で停止できません。

各ボタンについて、以下に説明します。

[Help] ボタン : ヘルプ画面を表示します。

[OK] ボタン : 設定変更を保存し、オプション画面を閉じます。

[Cancel] ボタン : 設定変更を保存せず、オプション画面を閉じます。

第 2 部 EmbedWare/ SysMon SDK

この章では、EmbedWare/SysMon SDK の概要について説明します。

1.1 EmbedWare/SysMon SDK とは

EmbedWare/SysMon SDK では、EmbedWare/SysMon Entry の機能をアプリケーションから利用するための API を提供しています。この API を利用することで、アプリケーションに容易にハードウェア監視機能を追加できます。

1.2 機能

EmbedWare/SysMon SDK には、以下の機能があります。

機能	説明
コンピューターの状態の取得	コンピューターの温度、電圧値、ファン回転数の異常の有無、現在の値、過去最高値・過去最低値、異常判定の際の上限および下限値を取得します。
S.M.A.R.T. 監視	内蔵ディスクデバイスの S.M.A.R.T. 機能による異常検出の有無、および S.M.A.R.T. アトリビュートデータを取得します。
RAID 監視	チップセット内蔵 RAID コントローラー、RAID カード（タイプ 1）、RAID カード（タイプ 2）、ソフトウェア RAID の状態を取得します。RAID カード（タイプ 2）とソフトウェア RAID は、接続ディスクの S.M.A.R.T. アトリビュートデータおよびモデル名、シリアル番号を取得できます。
システムエラーに関する情報の取得	PCI バス、メモリ、BMC エラーの有無、エラー発生時の BIOS イベントログデータを取得します。
パフォーマンス情報の取得	CPU の実周波数情報を取得します。
動作設定の取得	各種監視機能の有効・無効、異常発生時のメッセージボックス表示設定、Windows イベントログ設定を確認します。 温度、電圧値、ファン監視で異常検出時のシャットダウン設定を確認します。
ウォッチドッグタイマの操作	アプリケーションから直接ウォッチドッグタイマを制御します。
GPIO (General Purpose Input / Output) の操作	システム オン モジュールの GPIO を操作します。



RAID 監視で監視対象としている RAID カードは、2 種類存在します。

備考

- RAID カード（タイプ 1）は PCIe インターフェイスカードタイプです。
- RAID カード（タイプ 2）は SATA インターフェイスカードタイプです。

機種によって搭載される RAID カードは異なります。

搭載 RAID カードが不明な場合は、担当営業にご相談ください。



- GPIO 操作は、システム オン モジュールで使用可能です。エンベデッドコンピューターでは使用できません。
- 監視情報を取得する API は、過渡的な情報を取得する場合があります。
例えば、ESM_RAIDInfo() 関数で物理ディスクステータスが Rebuilding を示しているにもかかわらず、論理ボリュームステータスが Degraded を示すなどの場合があります。過渡的な情報が得られた場合は 5 秒後に再度 API を実行してください。

1.3 アプリケーション開発環境

EmbedWare/SysMon SDK を利用してアプリケーションを開発するための環境について、以下に説明します。

(1) インストール環境

EmbedWare/SysMon SDK のアプリケーション開発環境をインストールできる環境を以下に示します。

OS	: Windows 7 Professional SP1 以降、Windows 10 IoT Enterprise
ディスク資源	: 約 0.2MB



「システムの復元」が有効になっている場合は、「システムの復元」の設定に従い、ディスク資源が使用されます。

(2) アプリケーション作成時の留意事項

EmbedWare/SysMon SDK は 32bit および 64bit DLL が提供されています。実行する OS およびアプリケーションのアーキテクチャーによって使い分ける必要があります。

- アプリケーションのコンパイル・リンクには esm_api.h、esm_api.lib の 2 つのファイルが必要です。開発環境上で、インストールフォルダーからコピーして使用してください。
以下のファイルが提供されます。
 - esm_api.dll DLL ファイル
 - esm_api.h ヘッダファイル
 - esm_api.lib ライブラリファイル (DLL エントリーポイント)

1.4 アプリケーション実行環境

EmbedWare/SysMon SDK で開発したアプリケーションを実行するためには、EmbedWare/SysMon SDK ランタイムライブラリをターゲット装置にインストールする必要があります。

(1) インストール環境

EmbedWare/SysMon SDK ランタイムライブラリをインストールできる環境を以下に示します。

OS	: Windows 7 Professional SP1 以降、Windows 10 IoT Enterprise
ディスク資源	: 約 0.2MB
その他	: EmbedWare/SysMon Entry Ver 7.1 があらかじめインストールされている必要があります。



「システムの復元」が有効になっている場合は、「システムの復元」の設定に従い、ディスク資源が使用されます。



上記以外の環境をご使用のお客様は、担当営業にご相談ください。

(2) アプリケーション実行環境作成時の留意事項

- EmbedWare/SysMon SDK は 32bit および 64bit DLL が提供されています。実行する OS および アプリケーションのアーキテクチャーによって使い分ける必要があります。esm_api.dll がシステムフォルダーに配置されます。

第2章 インストールとアンインストール

この章では、EmbedWare/SysMon SDK のインストールとアンインストールの方法について説明します。

2.1 インストール

◇インストールする前に

EmbedWare/SysMon SDK では 32bit 版、64bit 版と 2 種類のパッケージを提供しています。

EmbedWare/SysMon SDK をインストールする前に、以下のことを確認してください。

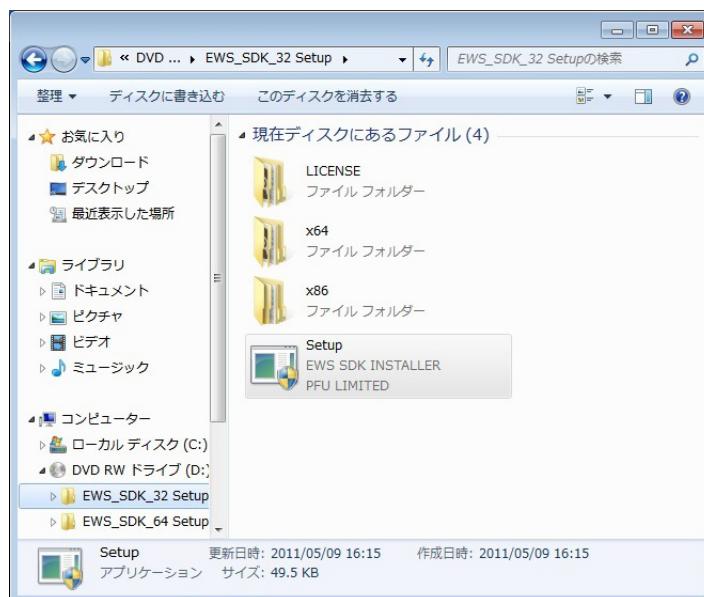
- Administrator 権限でログオンしている
- 実行中のアプリケーションがすべて終了している

◇アプリケーション開発環境をインストールする

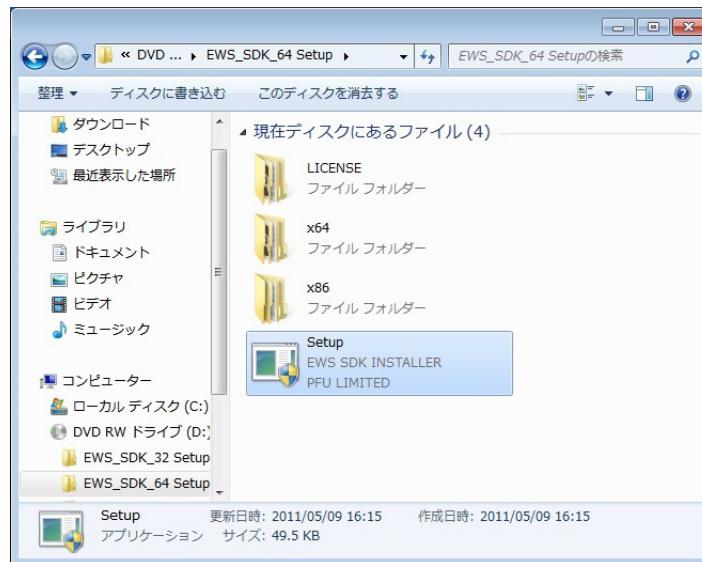
- 1 「EmbedWare/SysMon SDK Ver 7.1」 アプリケーション CD をセットします。
- 2 CD-ROM ドライブを選択し、「EWS_SDK_32 Setup」 または「EWS_SDK_64 Setup」 フォルダーをダブルクリックします。

フォルダーの内容が表示されます。

- 32bit 版の場合



- 64bit 版の場合



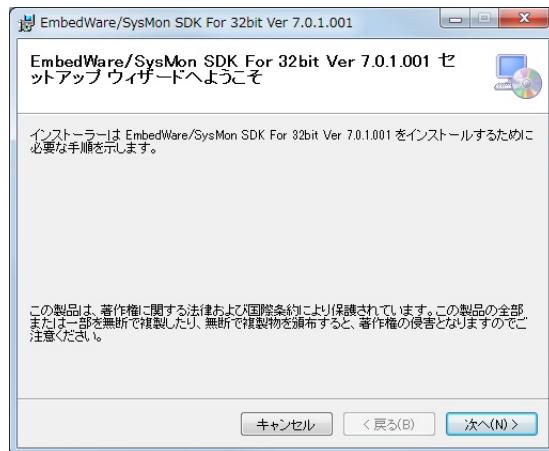
3 "Setup.exe" をダブルクリックします。

以下のメッセージが表示された場合、[はい] ボタンをクリックします。

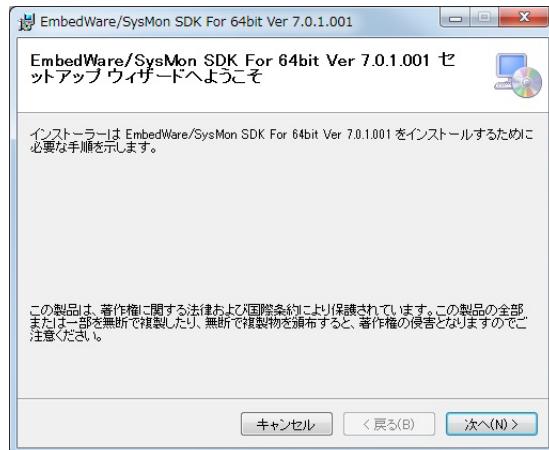


EmbedWare/SysMon SDK のインストーラーが起動されます。

- 32bit 版の場合



- 64bit 版の場合

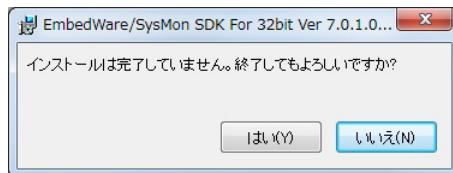




インストーラー画面の「キャンセル」ボタンをクリックすると、インストールを中断できます。

a [キャンセル] ボタンをクリックします。

- インストールの中止を確認するメッセージボックスが表示されます。
- 32bit 版の場合

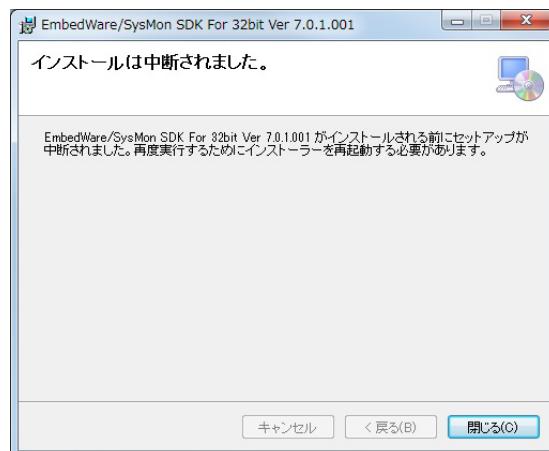


- 64bit 版の場合



b [はい] ボタンをクリックします。

- インストールが中断されたことを示す画面が表示されます。
- 32bit 版の場合





- 64bit 版の場合



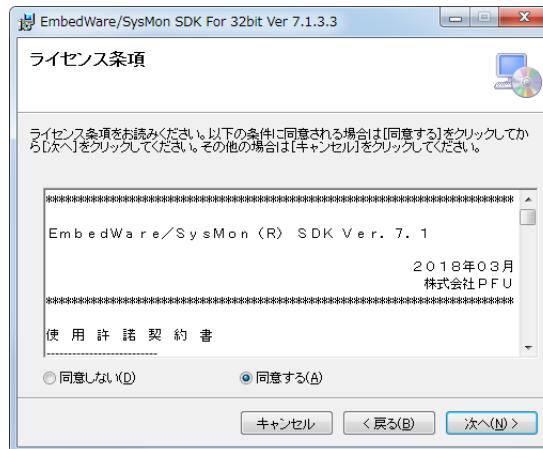
手順 a で [いいえ] ボタンをクリックすると、確認メッセージボックスが閉じられ、インストーラー画面に戻ります。

C [閉じる] ボタンをクリックします。

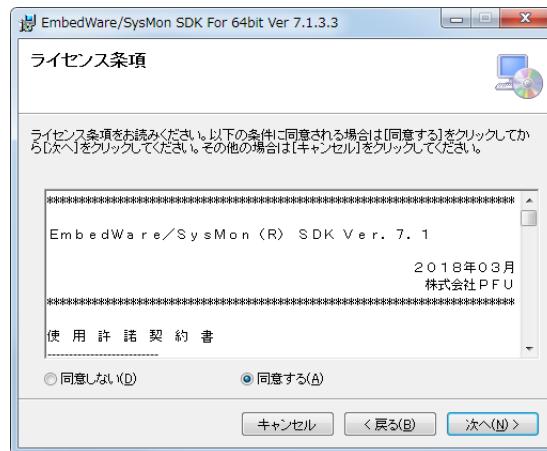
4 [次へ] ボタンをクリックします。

ライセンス条項画面が表示されます。

- 32bit 版の場合



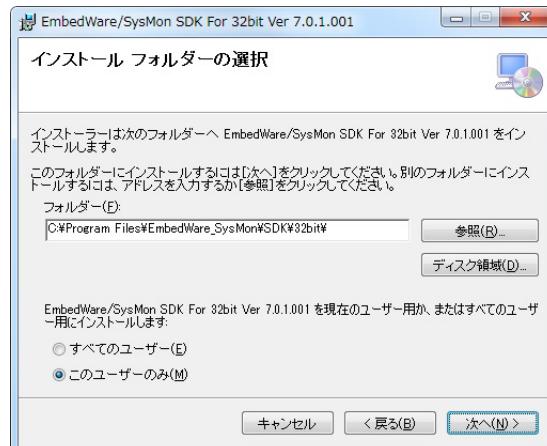
- 64bit 版の場合



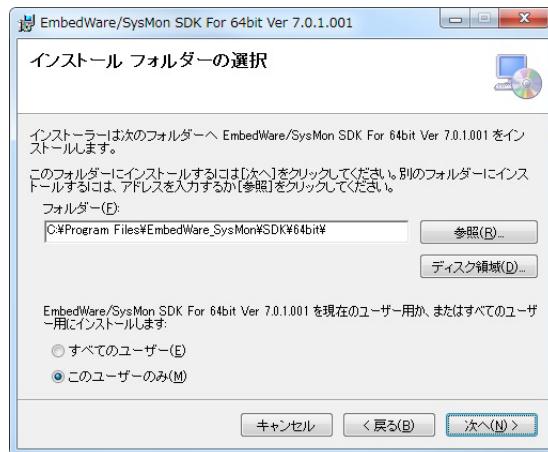
5 [同意する] ラジオボタンを選択し、[次へ] ボタンをクリックします。

インストールフォルダーの選択画面が表示されます。

- 32bit 版の場合

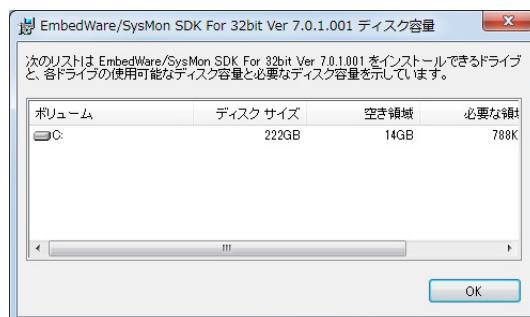


- 64bit 版の場合



インストール先を変更する場合は、[参照...] ボタンをクリックして変更します。
ディスク情報を確認する場合は、[ディスク領域...] ボタンをクリックして、ディスク情報画面を表示します。

- 32bit 版の場合



- 64bit 版の場合

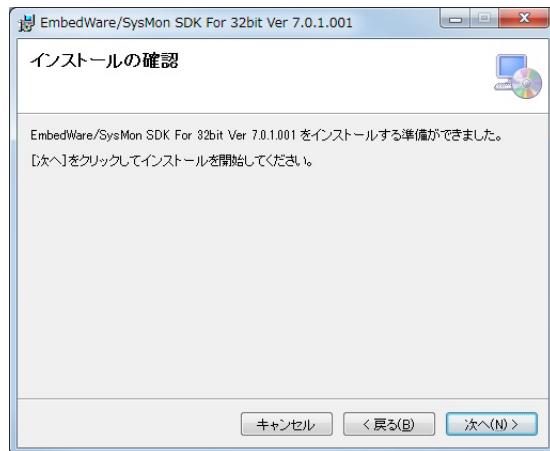


- ・ボリューム : 存在するディスクの名前が表示されます。
- ・ディスク サイズ : ディスクのサイズが表示されます。
- ・空き領域 : ディスクの空き領域が表示されます。
- ・必要な領域 : ディスクの使用領域が表示されます。
- ・残領域 : ディスクの残領域が表示されます。

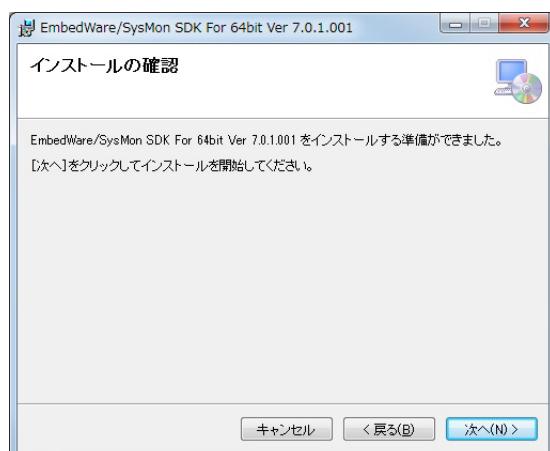
6 [次へ] ボタンをクリックします。

インストールの開始確認画面が表示されます。

- ・32bit 版の場合



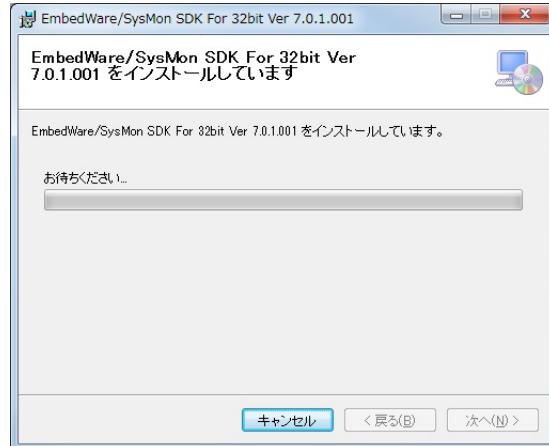
- ・64bit 版の場合



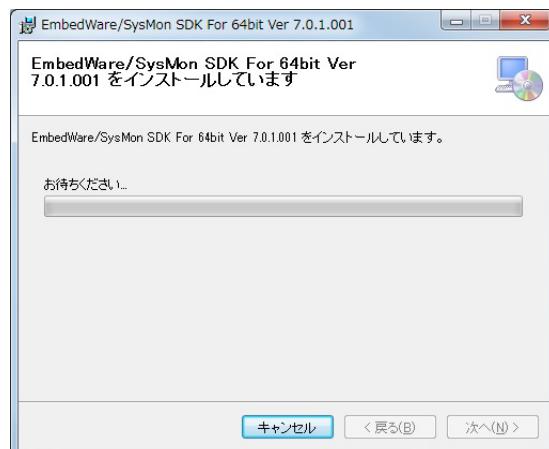
7 [次へ] ボタンをクリックします。

インストールが開始され、インストールの経過画面が表示されます。
インストールを中止する場合は、[キャンセル] ボタンをクリックします。

- 32bit 版の場合

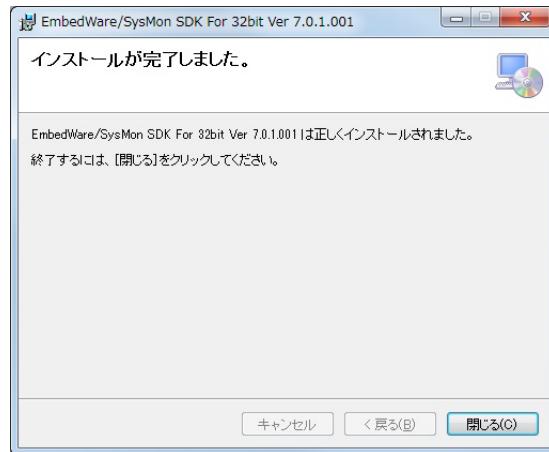


- 64bit 版の場合

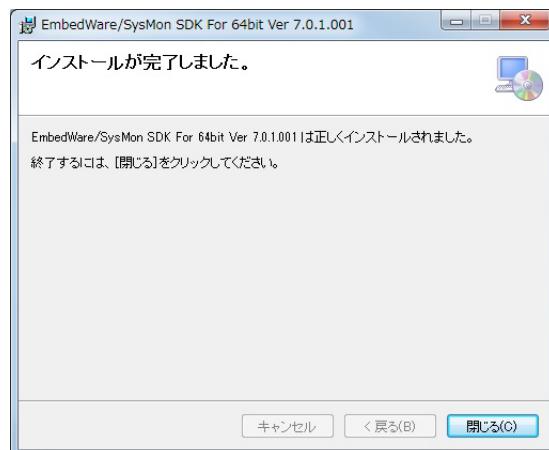


インストールが終わると、完了画面が表示されます。

- 32bit 版の場合



- 64bit 版の場合



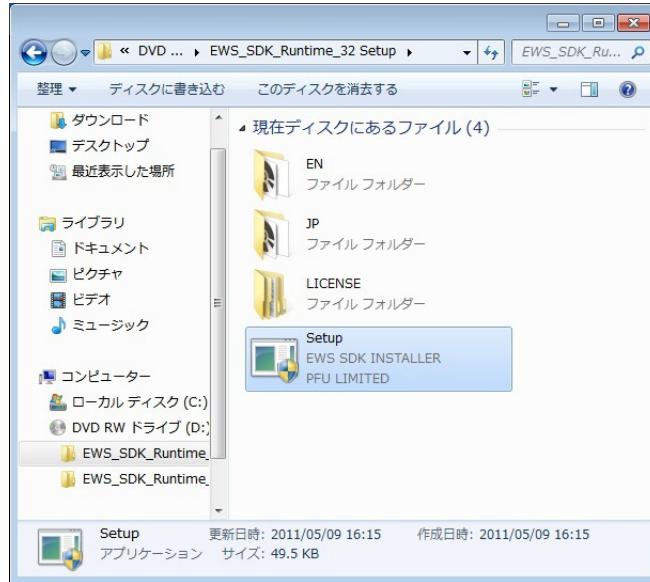
8 [閉じる] ボタンをクリックします。

EmbedWare/SysMon SDK アプリケーション開発環境のインストールが終了します。

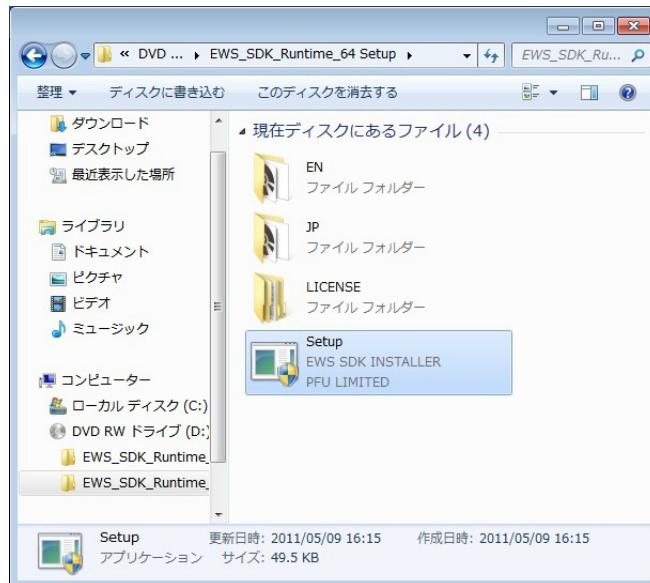
◇アプリケーション実行環境をインストールする

- 1 「EmbedWare/SysMon SDK Ver 7.1」 アプリケーション CD をセットします。
- 2 CD-ROM ドライブを選択し、「EWS_SDK_Runtime_32 Setup」 または 「EWS_SDK_Runtime_64 Setup」 フォルダーをダブルクリックします。
フォルダーの内容が表示されます。

- 32bit 版の場合



- 64bit 版の場合



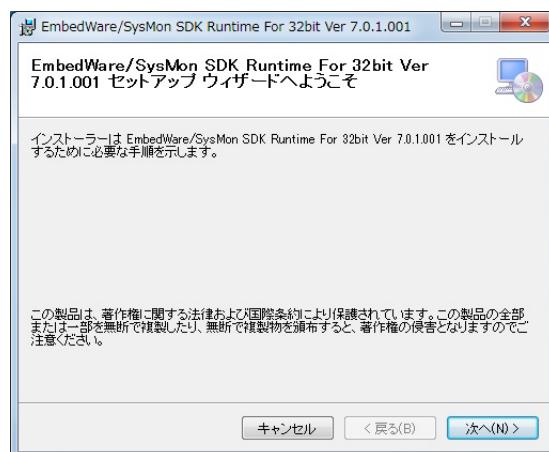
3 "Setup.exe" をダブルクリックします。

以下のメッセージが表示された場合、[はい] ボタンをクリックします。

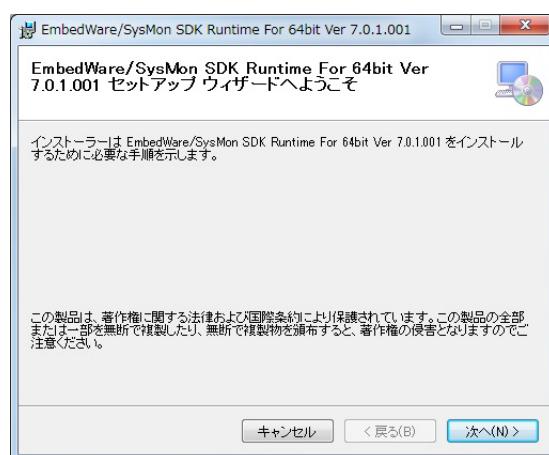


EmbedWare/SysMon SDK ランタイムのインストーラーが起動されます。

- 32bit 版の場合



- 64bit 版の場合





インストーラー画面の【キャンセル】ボタンをクリックすると、インストールを中断できます。

a [キャンセル] ボタンをクリックします。

インストールの中止を確認するメッセージボックスが表示されます。

- 32bit 版の場合



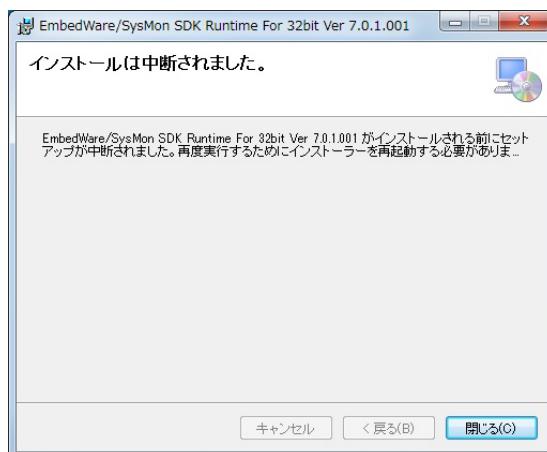
- 64bit 版の場合



b [はい] ボタンをクリックします。

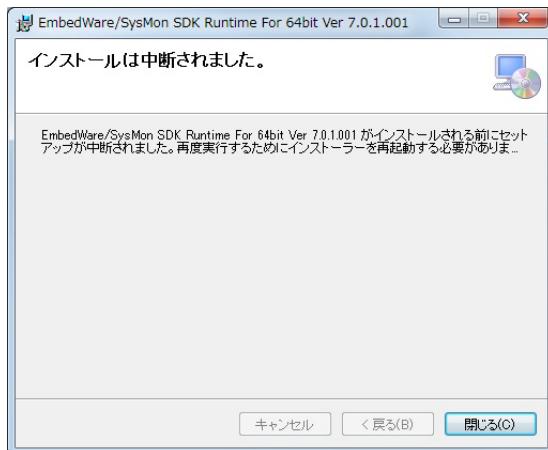
インストールが中断されたことを示す画面が表示されます。

- 32bit 版の場合





- 64bit 版の場合



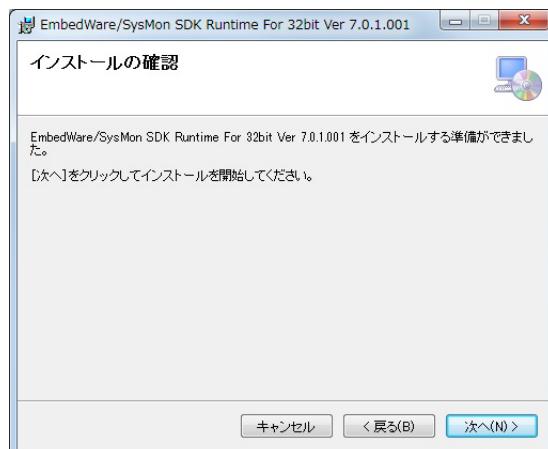
手順 a で [いいえ] ボタンをクリックすると、確認メッセージボックスが閉じられ、インストーラー画面に戻ります。

C [閉じる] ボタンをクリックします。

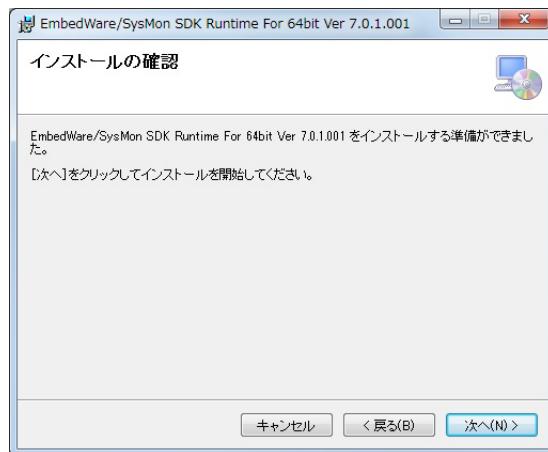
4 [次へ] ボタンをクリックします。

インストールの開始確認画面が表示されます。

- 32bit 版の場合



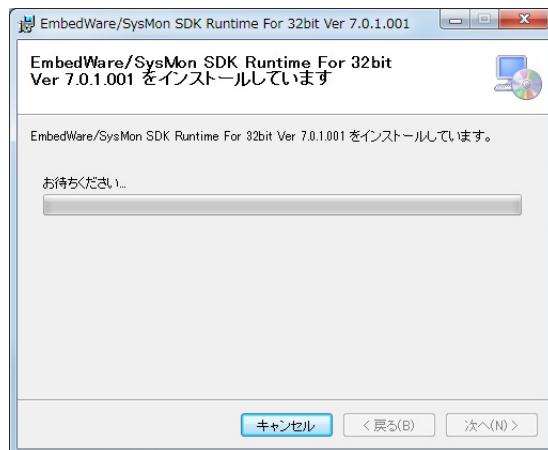
- 64bit 版の場合



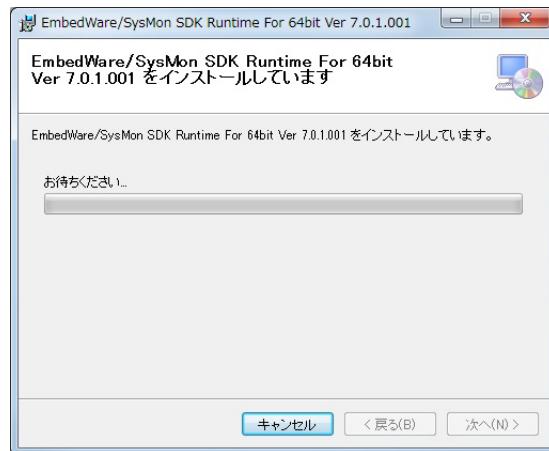
5 [次へ] ボタンをクリックします。

インストールが開始され、インストールの経過画面が表示されます。
インストールを中止する場合は、[キャンセル] ボタンをクリックします。

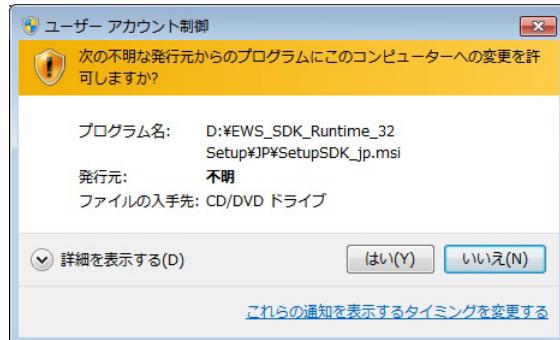
- 32bit 版の場合



- 64bit 版の場合

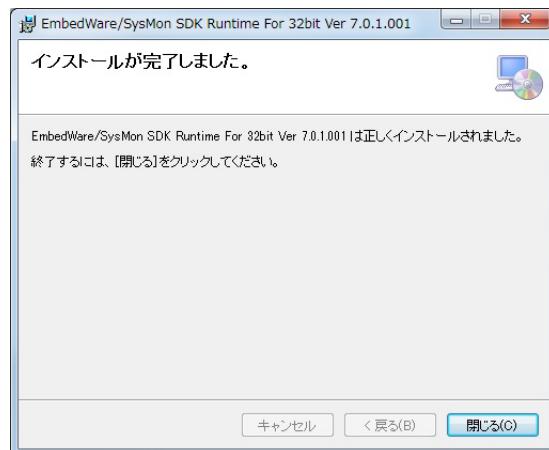


以下のメッセージが表示された場合、[はい] ボタンをクリックします。

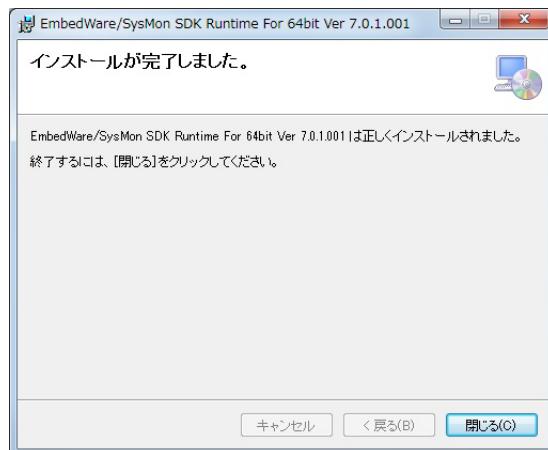


インストールの完了画面が表示されます。

- 32bit 版の場合



- 64bit 版の場合



6 [閉じる] ボタンをクリックします。

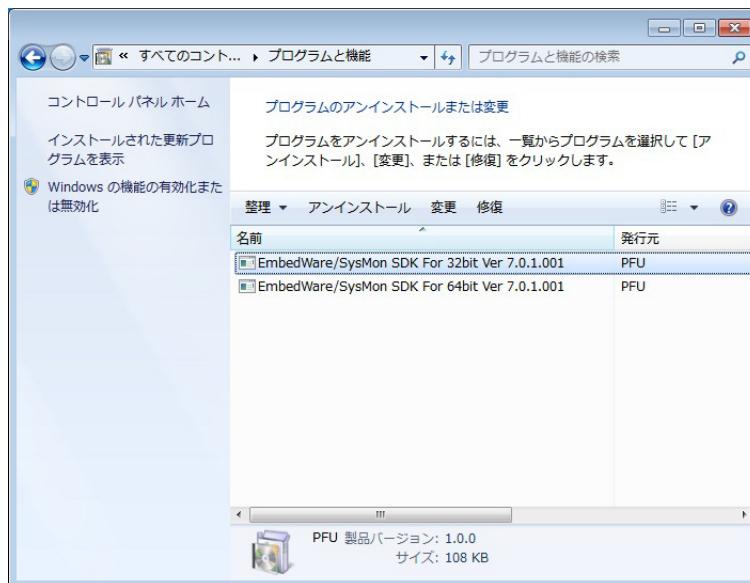
EmbedWare/SysMon SDK ランタイムライブラリのインストールが終了します。

2.2 アンインストール

◇アプリケーション開発環境をアンインストールする

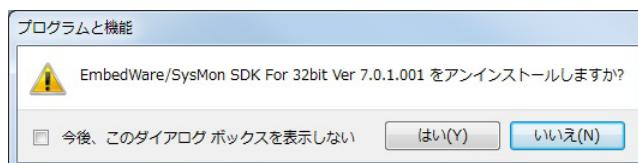
1 コントロールパネルから、EmbedWare/SysMon SDK の削除を起動します。

- 1 「スタート」 → 「コントロールパネル」をクリックします。
- 2 「プログラムのアンインストール」をクリックします。
- 3 「EmbedWare/SysMon SDK For 32bit」または「EmbedWare/SysMon SDK For 64bit」をダブルクリックします。



以下の確認メッセージが表示されます。

- 32bit 版の場合

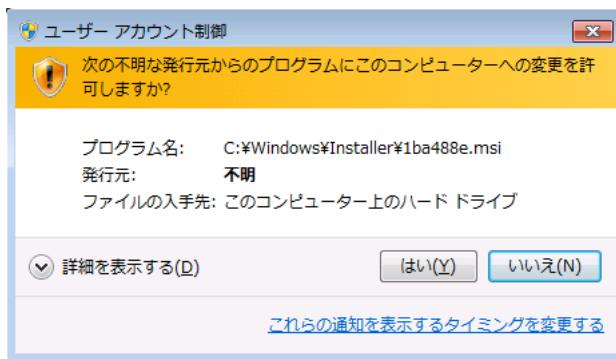


- 64bit 版の場合



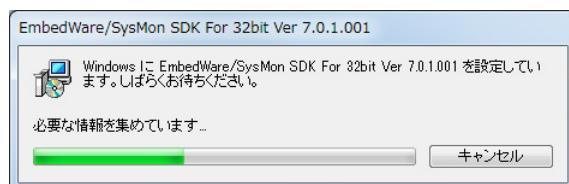
2 [はい] ボタンをクリックします。

以下のメッセージが表示された場合、[はい] ボタンをクリックします。

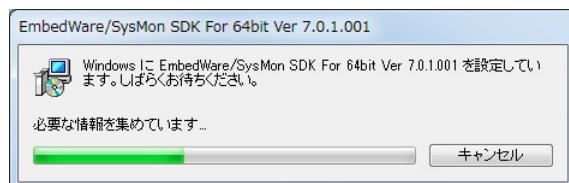


アンインストールが開始され、アンインストールの経過画面が表示されます。

- 32bit 版の場合



- 64bit 版の場合



アンインストールを中止すると、不具合が生じる場合があるため、[キャンセル] ボタンをクリックしないでください。

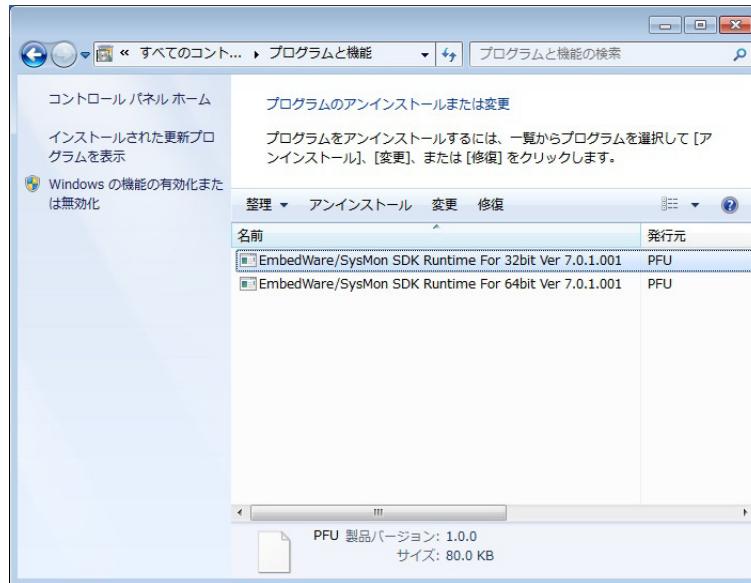
[キャンセル] ボタンをクリックして不具合が生じた場合は、「スタート」→「コントロールパネル」→「プログラムのアンインストール」から、「EmbeddedWare/SysMon SDK For 32bit」または「EmbeddedWare/SysMon SDK For 64bit」を選択し、「変更」ボタンをクリックして、プログラムの修復を行ってください。

それでも問題がある場合は、担当営業にご相談ください。

◇アプリケーション実行環境をアンインストールする

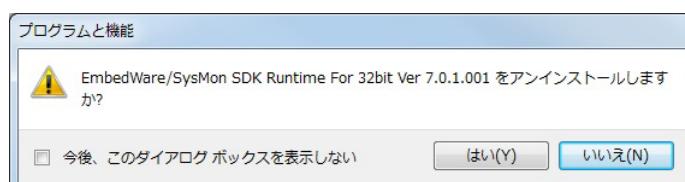
1 コントロールパネルから、EmbedWare/SysMon SDK Runtime の削除を起動します。

- 1 「スタート」 → 「コントロールパネル」をクリックします。
- 2 「プログラムのアンインストール」をクリックします。
- 3 「EmbedWare/SysMon SDK Runtime For 32bit」または「EmbedWare/SysMon SDK Runtime For 64bit」をダブルクリックします。



以下の確認メッセージが表示されます。

- 32bit 版の場合

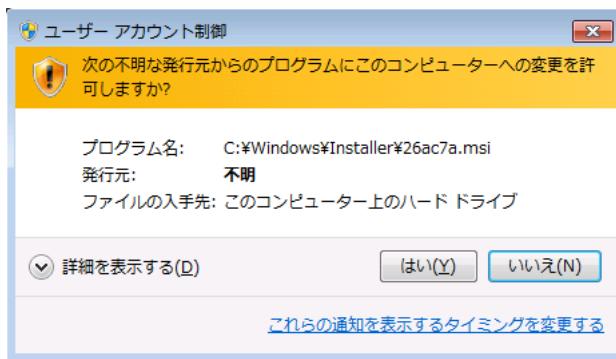


- 64bit 版の場合



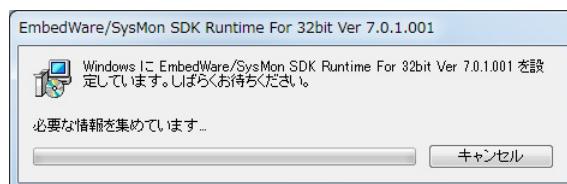
2 [はい] ボタンをクリックします。

以下のメッセージが表示された場合、[はい] ボタンをクリックします。

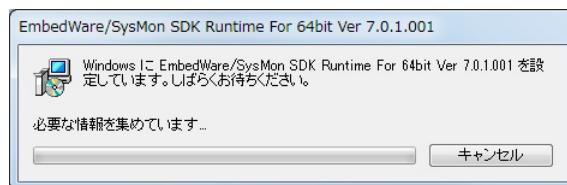


アンインストールが開始され、アンインストールの経過画面が表示されます。

- 32bit 版の場合



- 64bit 版の場合



アンインストールを中止すると、不具合が生じる場合があるため、[キャンセル] ボタンをクリックしないでください。

[キャンセル] ボタンをクリックして不具合が生じた場合は、「スタート」→「コントロールパネル」→「プログラムのアンインストール」から、「EmbedWare/SysMon SDK Runtime For 32bit」または「EmbedWare/SysMon SDK Runtime For 64bit」を選択し、[変更] ボタンをクリックして、プログラムの修復を行ってください。

それでも問題がある場合は、担当営業にご相談ください。

2.3 版数確認

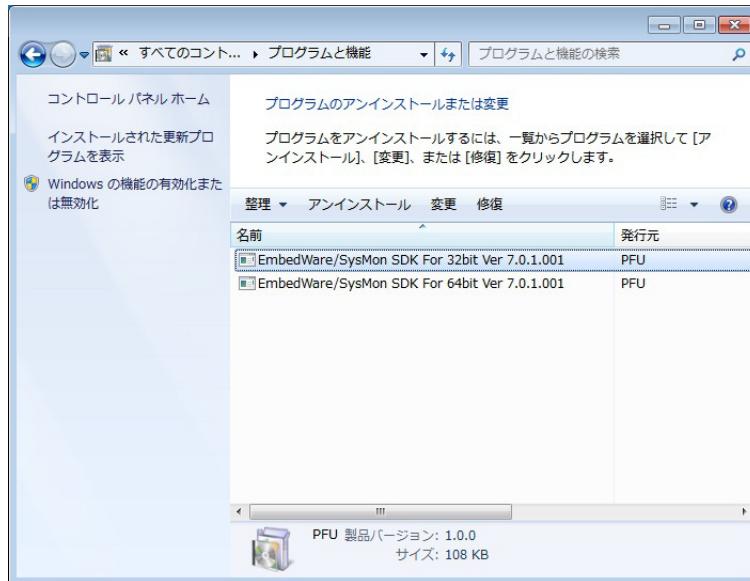
以下の手順で版数を確認します。

1 「スタート」→「コントロールパネル」をクリックします。

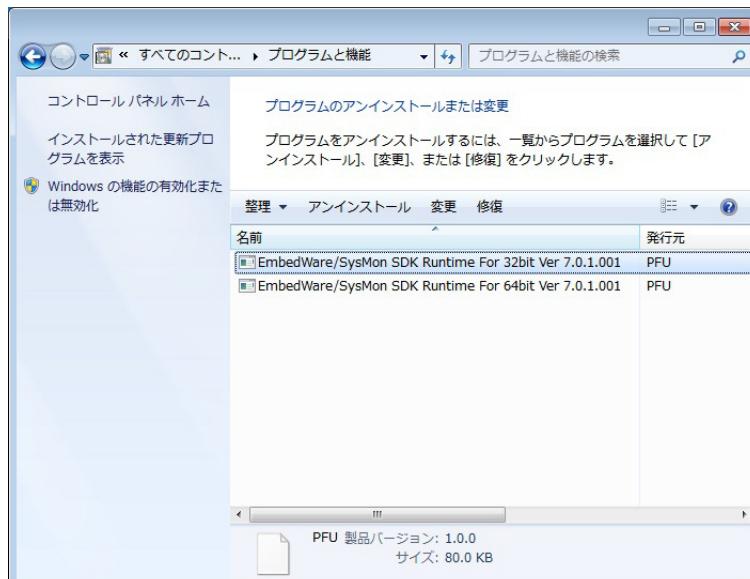
2 「プログラムのアンインストール」をクリックします。

版数が表示されます。

- アプリケーション開発環境の場合



- アプリケーション実行環境の場合



第3章 API仕様



この章では、EmbedWare/SysMon SDK を利用するにあたって、必要な API 仕様について説明します。

3.1 ESM_Open() 関数

3

API 仕様

機能説明

EmbedWare/SysMon Entry へのアクセスを開始します。

呼び出し形式

```
#include "esm_api.h"  
HANDLE ESM_Open(void);
```

パラメーター

なし

復帰値

ハンドル値	: 正常終了。EmbedWare/SysMon Entry へのアクセス開始を行うためのハンドルを返します。
ESM_ERROR(-1)	: システムエラーが発生。

使用例

```
#include "esm_api.h"  
HANDLE hesm; /* EmbedWare/SysMon アクセスハンドル */  
hesm = ESM_Open(); /* EmbedWare/SysMon アクセスハンドル獲得 */  
if ( hesm == (HANDLE)ESM_ERROR ) {  
    /* エラー処理 */  
}  
/* EmbedWare/SysMon アクセス要求開始 */  
:
```

3.2 ESM_Close() 関数

機能説明

EmbedWare/SysMon Entryへのアクセスを終了します。
ESM_Open()と対で必ず実行してください。

呼び出し形式

```
#include "esm_api.h"  
int ESM_Close(HANDLE hesm);
```

パラメーター

HANDLE hesm : ESM_Open()で獲得したハンドルを指定します。

復帰値

ESM_SUCCESS(0) : 正常終了。
ESM_ERROR(-1) : システムエラーが発生。

使用例

```
#include "esm_api.h"  
  
HANDLE hesm; /* EmbedWare/SysMon アクセスハンドル */  
hesm = ESM_Open(); /* EmbedWare/SysMon アクセスハンドル獲得 */  
if( hesm == (HANDLE)ESM_ERROR ) {  
    /* エラー処理 */  
}  
/* EmbedWare/SysMon アクセス要求開始 */  
:  
:  
/* EmbedWare/SysMon アクセス要求終了 */  
ESM_Close( hesm ); /* EmbedWare/SysMon アクセスハンドル解放 */  
:
```

3.3 ESM_Get() 関数

機能説明

指定した項目の現在の状態または設定内容を取得します。

呼び出し形式

```
#include "esm_api.h"
int ESM_Get(HANDLE hesm, char *Key1, char *Key2, char *Rtn_Value);
```

パラメーター

HANDLE hesm : ESM_Open() で獲得したハンドルを指定します。
 char *Key1 : 監視または設定項目を示すキーワードへのポインターを指定します。指定できる文字列は以下のとおりです。
 TEMP1 ~ TEMP48 : 個々の温度センサーに関する情報を取得します。
 TEMP1 ~ TEMP48 に対応する温度センサーは、EmbedWare/SysMon Entry で確認してください。番号は、表示順に対応しています。

VOLTAGE1 ~ VOLTAGE48 : 個々の電圧センサーに関する情報を取得します。

 VOLTAGE1 ~ VOLTAGE48 に対応する電圧センサーは、EmbedWare/SysMon Entry で確認してください。番号は、表示順に対応しています。

FAN1 ~ FAN48 : 個々のファンに関する情報を取得します。

 FAN1 ~ FAN48 に対応するファンは、EmbedWare/SysMon Entry で確認してください。番号は、表示順に対応しています。

SLOT0 ~ SLOT7 : S.M.A.R.T. 監視に関する情報を取得します。

-  SLOT0 ~ SLOT7 に対応する S.M.A.R.T. 監視情報は、EmbedWare/SysMon Entry で確認してください。
- EmbedWare/SysMon Ver7.0 以前に使用していた DISK1 ~ DISK8 も指定可能です。DISK1 ~ DISK8 の番号は、SLOT の番号に 1 を加えた値です。

UPTIME : 総通電時間を取得します。

TEMP : 温度監視全般に関する設定を取得します。

VOLTAGE : 電圧監視全般に関する設定を取得します。

FAN : ファン監視全般に関する設定を取得します。

RAID : チップセット RAID および RAID カード (タイプ 1) 監視の共通設定を取得します。

 共通設定は、チップセット RAID および RAID カード (タイプ 1) 監視の各設定の論理和になります。

CHIPSETRAID : チップセット RAID 監視の設定を取得します。

RAIDCARD : RAID カード (タイプ 1) 監視の設定を取得します。

SVRAID : RAID カード (タイプ 2) 監視の設定を取得します。

SWRAID : ソフトウェア RAID 監視の設定を取得します。

PCI : BIOS イベントログによる PCI エラー監視の設定を取得します。

MEMORY : BIOS イベントログによるメモリエラー監視の設定を取得します。

BMC : BIOS イベントログによる BMC エラー監視の設定を取得します。

ALL	: すべての監視項目に共通した情報を取得します。
char *Key2	: 取得したい情報種別を示すキーワードへのポインターを指定します。指定できる文字列は以下のとおりです。実際に指定できる値は、Key1 の内容に依存します。※「補足説明」参照
Name	: センサー名
Value	: 現在の値
Status	: 現在の状態
Threshold	: 現在設定されているしきい値
Worst	: 過去最高値・過去最低値の記録と発生日時
Action	: 監視の有効・無効設定または異常発生時の動作設定
Log	: Windows イベントログに関する設定
Hddinfo	: ハードディスクのモデル名、シリアル番号
char *Rtn_Value	: 指定されたキーワードに対応した情報を格納する領域へのポインターを指定します。アプリケーションは最低 64byte (終端ヌル文字を含む) の格納領域を用意する必要があります。

補足説明

Key1、Key2 で指定可能なパラメーターの組み合わせと、Rtn_Value に返されるデータを以下に示します。

Key1	Key2	Rtn_Value (*1)	Rtn_Value の説明
TEMP1 ~ TEMP48	Name	(センサー名)	EmbedWare/SysMon Entry の「Temperature」で表示されるセンサー名と同じ
	Value	(整数)	現在値 (*2)
	Status	NORMAL	正常
		ERROR	異常発生
	Threshold	(整数) , (整数)	上限しきい値、下限しきい値 下限しきい値が存在しない場合、右側の整数は省略されます。 例) 100,
	Action	ON	監視有効
		OFF	監視無効
	Worst	(整数) , (整数) , (YY/MM/DD hh:mm:ss) , (YY/MM/DD hh:mm:ss)	過去最高温度、最低温度、最高温度の記録日時、最低温度の記録日時 (*2)

Key1	Key2	Rtn_Value (*1)	Rtn_Value の説明
VOLTAGE1 ~ VOLTAGE48	Name	(センサー名)	EmbedWare/SysMon Entry の「Voltage」で表示されるセンサー名と同じ
	Value	(小数)	現在値 (出力形式: 小数第四位まで表示) (*2)
	Status	NORMAL	正常
		ERROR	異常発生
	Threshold	(小数) , (小数)	上限しきい値、下限しきい値 (出力形式: 小数第四位まで表示)
	Action	ON	監視有効
		OFF	監視無効
	Worst	(小数) , (小数) , (YY/MM/DD hh:mm:ss) , (YY/MM/DD hh:mm:ss)	過去最高電圧、最低電圧 (出力形式: 小数第四位まで表示)、最高電圧の記録日時、最低電圧の記録日時 (*2)
FAN1 ~ FAN48	Name	(センサー名)	EmbedWare/SysMon Entry の「Fan Speed」で表示されるセンサー名と同じ
	Value	(整数)	現在値 (*2)
	Status	NORMAL	正常
		ERROR	異常発生
	Threshold	(整数) , (整数)	上限しきい値、下限しきい値 上限しきい値が存在しない場合、左側の整数は省略されます。 例) ,1000
	Action	ON	監視有効
		OFF	監視無効
	Worst	(整数) , (整数) , (YY/MM/DD hh:mm:ss) , (YY/MM/DD hh:mm:ss)	過去最高回転数、最低回転数、最高回転数の記録日時、最低回転数の記録日時 (*2)

Key1	Key2	Rtn_Value (*1)	Rtn_Value の説明
SLOT0 ~ SLOT7 DISK1 ~ DISK8	Name	(監視項目名)	EmbedWare/SysMon Entry の「S.M.A.R.T.」で表示されるセンサー名と同じ
	Status	OK	正常
		WARNING	異常発生
		UNKNOWN	S.M.A.R.T. ステータスが取得できない
	Hddinfo	(Model), (SerialNumber)	ハードディスクのモデル名、シリアル番号
UPTIME	Action	ON	監視有効
		OFF	監視無効
TEMP	Value	(小数)	総通電時間（単位：時間、出力形式：小数第一位まで表示）
VOLTAGE	Action	ON, (整数：0 ~ 60)	温度異常発生時シャットダウンする
		OFF	温度異常発生時シャットダウンしない
FAN	Action	ON, (整数：0 ~ 60)	ファン異常発生時シャットダウンする
		OFF	ファン異常発生時シャットダウンしない
RAID (チップセット RAID および RAID カード (タイプ 1) 監視の共 通設定)	Action	ON	監視有効
		OFF	監視無効
CHIPSETRAID	Action	ON	監視有効
		OFF	監視無効
RAIDCARD	Action	ON	監視有効
		OFF	監視無効
SVRAID	Action	ON	監視有効
		OFF	監視無効
SWRAID	Action	ON	監視有効
		OFF	監視無効

Key1	Key2	Rtn_Value (*1)	Rtn_Value の説明
PCI (PCI エラー監視)	Action	ON	監視有効
		OFF	監視無効
MEMORY (メモリエラー監視)	Action	ON	監視有効
		OFF	監視無効
BMC (BMC エラー監視)	Action	ON	監視有効
		OFF	監視無効
ALL	Action	ON, LONG	エラー発生時メッセージボックスを表示し、[OK] ボタンをクリックするまでビープ音を鳴らす
		ON, SHORT	エラー発生時メッセージボックスを表示し、3 秒間ビープ音を鳴らす
		OFF	エラー発生時メッセージボックスを表示しない
	Log	ON	エラー発生時 Windows イベントログ格納する
		OFF	エラー発生時 Windows イベントログ格納しない

*1 : 括弧付きのものは、値の内容を示します。括弧なしのものは、その値がそのまま格納されることを示します。

*2 : センサー非搭載時または監視準備中は、値が空になります。センサー搭載しているにもかかわらず値が空の場合は、監視準備中の可能性がありますので再取得してください。

復帰値

- ESM_SUCCESS(0) : 正常終了。
- ESM_ERROR(-1) : システムエラーが発生。
- ESM_PRMERR(-2) : 指定パラメーターに誤りがある。
- ESM_NODATA1(-9) : Key1 に誤りがある。
- ESM_NODATA2(-10) : Key1、Key2 の組み合わせに誤りがある。

使用例

```
#include "esm_api.h"
int rtn;
char rtnstr[64];
HANDLE hesm;      /* EmbedWare/SysMon アクセスハンドル */
hesm = ESM_Open( ); /* EmbedWare/SysMon アクセスハンドル獲得 */
if( hesm == (HANDLE)ESM_ERROR ) {
    /* エラー処理 */
}
/* EmbedWare/SysMon アクセス要求開始 */
:
memset( rtnstr, 0, 64 ); /* 獲得域の初期化 */
rtn = ESM_Get( hesm, "TEMP1", "Value", rtnstr ); /* TEMP1 の現在値獲得 */
if( rtn < 0 ) {
    /* エラー処理 */
}
/* 空文字（センサー非搭載または監視準備中）でない場合に表示する */
if( strcmp( "", rtnstr ) != 0 ) {
    printf( "TEMP1 = %s°C", rtnstr );
}
:
/* 次の監視項目へ */
:
/* EmbedWare/SysMon アクセス要求終了 */
ESM_Close( hesm ); /* EmbedWare/SysMon アクセスハンドル解放 */
```

使用例をコンソールアプリケーションとして作成し実行した場合、以下のように出力されます。

TEMP1 = 40°C

3.4 ESM_GetSmart() 関数

機能説明

指定したディスクの S.M.A.R.T. ステータスと S.M.A.R.T. アトリビュートデータを取得します。本関数には拡張関数が用意されており、「3.16 ESM_GetSmartEx() 関数」(95 ページ) の使用を推奨しています。

呼び出し形式

```
#include "esm_api.h"
int ESM_GetSmart(HANDLE hesm, char *Key1, smart_attributes_t *Get_Value);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
char *Key1	: 監視または設定項目を示すキーワードへのポインターを指定します。指定できる文字列は以下のとおりです。
DISK1 ~ DISK8	: 指定されたディスクに関する情報を取得します。
smart_attributes_t *Get_Value	: 情報を格納する領域へのポインター。以下に示す smart_attributes_t 構造体へのポインターを指定します。

smart_attr_t, smart_attributes_t 構造体

```
typedef struct smart_attr {
    unsigned char id;
    unsigned char flags[2];
    unsigned char value;
    unsigned char worst;
    unsigned char threshold;
    unsigned char raw[6];
} smart_attr_t;

typedef struct smart_attributes {
    int status;
    smart_attr_t attribute[30];
} smart_attributes_t;
```

補足説明

smart_attributes_t 構造体に格納される status の意味は、以下のとおりです。

- | | |
|---|--|
| 0 | : S.M.A.R.T. アトリビュートデータが取得できません（この場合 attribute の格納は行われません） |
| 1 | : S.M.A.R.T. ステータスは正常です |
| 2 | : S.M.A.R.T. ステータスの異常が検出されています |

復帰値

- | | |
|----------------|-------------------|
| ESM_SUCCESS(0) | : 正常終了。 |
| ESM_ERROR(-1) | : システムエラーが発生。 |
| ESM_PRMERR(-2) | : 指定パラメーターに誤りがある。 |

3.5 ESM_RAIDInfo() 関数

機能説明

チップセット RAID および RAID カード (タイプ 1) のステータス情報を取得します。

呼び出し形式

```
#include "esm_api.h"
int ESM_RAIDInfo(HANDLE hesm, int Controller, int Port, unsigned int *status, char *raidlevel);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
int Controller	: RAID コントローラーの種類を指定します。
0	: チップセット RAID の情報を取得します。
1	: RAID カード (タイプ 1) の情報を取得します。
int Port	: チップセット RAID、RAID カード構成情報を指定します。 チップセット RAID、RAID カード構成情報はデバイス番号、コネクター番号、コントローラー番号の論理和で表現されます。
0x000000XX : XX	: デバイス番号。チップセット RAID はポート番号、RAID カードはデバイス番号に対応しています。
0x0000XX00 : XX	: コネクター番号。チップセット RAID は 00 固定、RAID カードはコネクター番号に対応しています。
0x00XX0000 : XX	: コントローラー番号。チップセット RAID は 00 固定、RAID カードはコントローラー番号に対応しています。ここで指定するコントローラー番号の開始番号は 00 です。EmbedWare/SysMon Entry の表示より 1 小さいのでご注意ください。
unsigned int *status	: ステータス情報を格納する領域へのポインターを指定します。 ステータス情報は、以下の物理ディスクステータス、物理ディスク S.M.A.R.T. ステータス、論理ボリュームステータス、構成情報の論理和で表現されます。
物理ディスクステータス	
None : 0x00	: パラメーター Port で指定したディスクが存在しない
Online : 0x01	: RAID 構成されているディスク
HotSpare : 0x02	: ホットスペアディスク
Failed : 0x04	: ディスクの異常が検出された
Rebuilding : 0x08	: リビルド中
NoArray : 0x10	: RAID 構築されていません
物理ディスク S.M.A.R.T. ステータス	
NORMAL : 0x00	: ディスクの S.M.A.R.T. 異常が検出されなかった (ディスクの S.M.A.R.T. 状態が正常または不明)
ERROR : 0x80	: ディスクの S.M.A.R.T. 異常が検出された
論理ボリュームステータス	
None/NoArray : 0x0000	: 物理ディスクが RAID 構築されていないか、ディスクが存在しない
OK : 0x0100	: 論理ボリュームが正常
Degraded : 0x0200	: 論理ボリュームの冗長性が失われている
Rebuilding : 0x0400	: 論理ボリュームがリビルド中
Failed : 0x0800	: 論理ボリュームが異常

構成情報

0xXXXX0000 : XXXX : 論理ボリューム番号
 char *raidlevel : RAID レベル情報を格納する領域へのポインターを指定します。
 この情報が不要な場合は NULL を指定します。指定した領域に RAID レベル情報を "0"、"1"、"5"、"10" などの文字列で格納します。領域のサイズは終端ヌル文字を含め最低 8 バイト準備する必要があります。指定した場所が RAID 構成ではない場合、長さ 0 の文字列が格納されます。

復帰値

ESM_SUCCESS(0) : 正常終了。
 ESM_ERROR(-1) : システムエラーが発生。
 ESM_PRMERR(-2) : 指定パラメーターに誤りがある。

留意事項

監視が有効ではない場合（装置が RAID 構成ではない、または、監視を無効に設定している）でもこの関数がエラーとなることはありませんが、すべてのステータス情報は 0 となります。

3.6 ESM_SvRAIDInfo() 関数

機能説明

RAIDカード（タイプ2）のステータス情報を取得します。

呼び出し形式

```
#include "esm_api.h"
int ESM_SvRAIDInfo(HANDLE hesm, int Array, int Drive, unsigned int *status, char *raidlevel);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
int Array	: 論理ボリュームの番号を指定します。0～3の値が指定可能です。
int Drive	: ディスク番号を指定します。0～1の値が指定可能です。
unsigned int *status	: ステータス情報を格納する領域へのポインターを指定します。 ステータス情報は、下記の物理ディスクステータス、物理ディスク S.M.A.R.T. ステータス、論理ボリュームステータス、構成情報の論理和で表現されます。

物理ディスクステータス

None : 0x00	: RAIDカードが検出できなかった
Online : 0x01	: RAID構成されているディスク
Failed : 0x02	: ディスクの異常が検出された
Rebuilding : 0x04	: リビルド中

物理ディスク S.M.A.R.T. ステータス

NORMAL : 0x00	: ディスクの S.M.A.R.T. 異常が検出されなかった（ディスクの S.M.A.R.T. 状態が正常または不明）
ERROR : 0x80	: ディスクの S.M.A.R.T. 異常が検出された

論理ボリュームステータス

None : 0x0000	: 物理ディスクが RAID構築されていないか、ディスクが存在しない
OK : 0x0100	: 論理ボリュームが正常
Degraded : 0x0200	: 論理ボリュームの冗長性が失われている
Rebuilding : 0x0400	: 論理ボリュームがリビルド中
Failed : 0x0800	: 論理ボリュームが異常

構成情報

0xXXXX0000	: XXXX : 論理ボリューム番号
char *raidlevel	: RAIDレベル情報を格納する領域へのポインターを指定します。 この情報が不要な場合は NULLを指定します。指定した領域に RAIDレベル情報を "0"、"1"、"5"、"10"などの文字列で格納します。領域のサイズは終端ヌル文字を含め最低8バイト準備する必要があります。

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。

留意事項

監視が有効ではない場合（RAID カードが存在しない、または、監視を無効に設定している）でもこの関数がエラーとなることはありませんが、すべてのステータス情報は 0 となります。

3.7 ESM_QueryError() 関数

機能説明

PCI エラー、メモリエラー、BMC エラー情報を取得します。

呼び出し形式

```
#include "esm_api.h"
int ESM_QueryError(HANDLE hesm, char *Key1, int *status, char *detail_info);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
char *Key1	: 監視または設定項目を示すキーワードへのポインターを指定します。指定できる文字列は以下のとおりです。
PCI	: PCI エラー情報を取得します。
MEMORY	: メモリエラー情報を取得します。
BMC	: BMC エラー情報を取得します。
int *status	: ステータス情報を格納する領域へのポインターを指定します。格納される値は以下のとおりです。
0	: 異常は検出されていません。
1	: 異常が検出されました。
char *detail_info	: 詳細情報を格納する領域へのポインターを指定します。上記 Status に 1 が返された場合に、この領域に値が格納されます。異常検出時の BIOS イベントログのバイナリデータを格納します。最低 256byte の領域を準備する必要があります。この情報が不要な場合は NULL を指定します。

詳細情報の格納形式は以下のとおりです。

Byte1	: イベントログタイプ
Byte2	: Byte1 ~ 8 を含むログ全体の長さを示す 7bit の値が格納されます（最上位ビットはリザーブです）
Byte3	: タイムスタンプ（年）
Byte4	: タイムスタンプ（月）
Byte5	: タイムスタンプ（日）
Byte6	: タイムスタンプ（時）
Byte7	: タイムスタンプ（分）
Byte8	: タイムスタンプ（秒）
Byte9 ~ 有効レコード長	: ログ固有のデータ領域

復帰値

- ESM_SUCCESS(0) : 正常終了。
- ESM_ERROR(-1) : システムエラーが発生。
- ESM_PRMERR(-2) : 指定パラメーターに誤りがある。

3.8 ESM_GetWDTTimeOut() 関数

機能説明

ウォッチドッグタイマのタイムアウト時間に設定可能な最大値（秒）、最小値（秒）を取得します。

呼び出し形式

```
#include "esm_api.h"  
int ESM_GetWDTTimeOut(HANDLE hesm, int *Max_Time, int *Min_Time);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
int *Max_Time	: ウォッチドッグタイマのタイムアウト時間に設定可能な最大値（秒）を格納する領域へのポインターを指定します。
int *Min_Time	: ウォッチドッグタイマのタイムアウト時間に設定可能な最小値（秒）を格納する領域へのポインターを指定します。

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。

3.9 ESM_SwitchWDT() 関数

機能説明

SDK のウォッチドッグタイマのオン／オフを設定します。EmbedWare/SysMon Entry のウォッチドッグタイマがオフの場合に設定できます。

本関数と「3.10 ESM_WDTHeartbeat() 関数」(92 ページ) は同一プロセスで実行する必要があります。ウォッチドッグタイマをオンに設定し、タイムアウト時間が経過した場合、指定したアクション動作が実行されます。

異なるプロセスでウォッチドッグタイマを操作した場合、動作は保証されません。

呼び出し形式

```
#include "esm_api.h"
int ESM_SwitchWDT(HANDLE hesm, int time, int action, int status);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
int time	: ウォッチドッグタイマのタイムアウト時間 (秒) を指定します。 タイムアウト時間の指定範囲は ESM_GetWDTTimeOut() から取得します。
int action	: ウォッチドッグタイマのタイムアウト発生時のアクションを指定します。指定できる値は以下のとおりです。 0 : 装置を再起動します (Hard Reset)。 1 : 何も実施しません (No Action)。 2 : 装置をシャットダウンします (Power Down)。 3 : 装置をシャットダウンしたあとに起動します (Power Cycle)。
	装置でサポートしているアクションは EmbedWare/SysMon Entry で確認してください。 詳細は、「3.14 ウォッチドッグタイマ設定」(44 ページ) を参照してください。

備考

int status	: ウォッチドッグタイマのオン／オフを設定します。指定できる値は以下のとおりです。 0 : オフ 1 : オン
------------	---

注意

ウォッチドッグタイマがすでにオンの状態でタイムアウト時間およびタイムアウト発生時のアクションを変更する場合は、一度ウォッチドッグタイマをオフにしてください。

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。
ESM_NOACCESS(-8)	: Administrator アクセス権限がない。
ESM_WDTERR1(-11)	: EmbedWare/SysMon Entry のウォッチドッグタイマ設定がオンである。
ESM_NOSUPPORT(-13)	: サポートしていないアクションが指定された。

3.10 ESM_WDTHeartbeat() 関数

機能説明

ウォッチドッグタイマのカウントをリセットします。SDKのウォッチドッグタイマがオンの場合に有効です。

本関数と「3.9 ESM_SwitchWDT() 関数」(91 ページ) は同一プロセスで実行する必要があります。異なるプロセスでウォッチドッグタイマを操作した場合、動作は保証されません。

呼び出し形式

```
#include "esm_api.h"
int ESM_WDTHeartbeat(HANDLE hesm);
```

パラメーター

HANDLE hesm : ESM_Open() で獲得したハンドルを指定します。

復帰値

ESM_SUCCESS(0)	： 正常終了。
ESM_ERROR(-1)	： システムエラーが発生。
ESM_PRMERR(-2)	： 指定パラメーターに誤りがある。
ESM_NOACCESS(-8)	： Administrator アクセス権限がない。
ESM_WDTERR2(-12)	： EWS_SwitchWDT() 関数を使用してウォッチドッグタイマをオンにしていない。

3.11 ESM_ReadGPI() 関数

機能説明

GPI (General Purpose Input) の情報を取得します。

呼び出し形式

```
#include "esm_api.h"
int ESM_ReadGPI(HANDLE hesm, BYTE *Read_Value);
```

パラメーター

HANDLE hesm	： ESM_Open() で獲得したハンドルを指定します。
BYTE *Read_Value	： GPI のデータを格納する領域へのポインターを指定します。 データの有効範囲は bit0~3 の 4 ビット。

復帰値

ESM_SUCCESS(0)	： 正常終了。
ESM_ERROR(-1)	： システムエラーが発生。
ESM_PRMERR(-2)	： 指定パラメーターに誤りがある。

3.12 ESM_ReadGPO() 関数

機能説明

GPO (General Purpose Output) の情報を取得します。

呼び出し形式

```
#include "esm_api.h"
int ESM_ReadGPO(HANDLE hesm, BYTE *Read_Value);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
BYTE *Read_Value	: GPO のデータを格納する領域へのポインターを指定します。 データの有効範囲は bit0 ~ 3 の 4 ビット。

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。

3.13 ESM_WriteGPO() 関数

機能説明

GPO に情報を設定します。

呼び出し形式

```
#include "esm_api.h"
int ESM_WriteGPO(HANDLE hesm, BYTE Write_Value);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
BYTE Write_Value	: GPO に設定するデータを指定します。データの有効範囲は bit0 ~ 3 の 4 ビット。有効範囲の各ビットに 0 または 1 を指定します。

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。
ESM_NOACCESS(-8)	: Administrator アクセス権限がない。

3.14 ESM_GetCpuCount() 関数

機能説明

物理CPU数を取得します。物理CPUの数を元に、「3.15 ESM_GetCpuPerform() 関数」(94ページ)を利用することでCPUの実周波数情報が取得できます。

呼び出し形式

```
#include "esm_api.h"
int ESM_GetCpuCount(HANDLE hesm, int *cpu_num);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
int *cpu_num	: 物理CPU数を格納する領域へのポインターを指定します。

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。

3.15 ESM_GetCpuPerform() 関数

機能説明

CPUの実周波数を取得します。

呼び出し形式

```
#include "esm_api.h"
int ESM_GetCpuPerform(HANDLE hesm, int slot, int *Get_Value);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
int slot	: 実周波数を取得するCPUの物理CPU番号(0～)を指定します。物理CPUの数は ESM_GetCpuCount() で取得します。
int *Get_Value	: CPU実周波数(MHz)の情報を格納する領域へのポインターを指定します。

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。
ESM_NODATA1(-9)	: CPUの実周波数情報が存在しない。

3.16 ESM_GetSmartEx() 関数

機能説明

指定したディスクの S.M.A.R.T. ステータスと S.M.A.R.T. アトリビュートデータを取得します。
ESM_GetSmartEx() は ESM_GetSmart() の拡張関数です。取得できる S.M.A.R.T. アトリビュートデータに違いがあります。

書式

```
#include "esm_api.h"
int ESM_GetSmartEx(HANDLE hesm, char *Key1, smart_data_t *Get_Value);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
char *Key1	: 監視または設定項目を示すキーワードへのポインターを指定します。指定できる文字列は以下のとおりです。
SLOT0 ~ SLOT7	: 指定されたディスクに関する情報を取得します。
smart_data_t *GetValue	: 情報を格納する領域へのポインター。以下に示す smart_data_t 構造体へのポインターを指定します。

smart_attribute_t, smart_data_t 構造体

```
typedef struct smart_attribute {
    unsigned char id;
    unsigned char flags[2];
    unsigned char value;
    unsigned char worst;
    unsigned char threshold;
    unsigned char raw[7];
    unsigned char reserved[3];
} smart_attribute_t;
```

```
typedef struct smart_data {
    int status;
    smart_attribute_t attributes[30];
} smart_data_t;
```



ESM_GetSmart() で扱う smart_attr_t, smart_attributes_t 構造体との差異は、以下のとおりです。

- ESM_GetSmart() と ESM_GetSmartEx() では、raw データおよび flags データの並びが違います。
- ESM_GetSmart() には raw データのうち 6 バイト分が格納されますが、ESM_GetSmartEx() には 7 バイト分すべてが格納されます。

flags データ Offset (ディスクから取得した時点での並び)	+0	+1
ESM_GetSmart()	flags[1]	flags[0]
ESM_GetSmartEx()	flags[0]	flags[1]

raw データ Offset (ディスクから取得した時 点での並び)	+0	+1	+2	+3	+4	+5	+6
ESM_GetSmart()	raw[5]	raw[4]	raw[3]	raw[2]	raw[1]	raw[0]	-
ESM_GetSmartEx()	raw[0]	raw[1]	raw[2]	raw[3]	raw[4]	raw[5]	raw[6]

補足説明

smart_data_t 構造体に格納される status の意味は、以下のとおりです。

- 0 : S.M.A.R.T. アトリビュートデータが取得できません
(この場合だけ attributes の格納は行われません)。
- 1 : S.M.A.R.T. ステータスは正常です。
- 2 : S.M.A.R.T. ステータスの異常が検出されています。

復帰値

- ESM_SUCCESS(0) : 正常終了。
- ESM_ERROR(-1) : システムエラーが発生。
- ESM_PRMERR(-2) : 指定パラメーターに誤りがある。

3.17 ESM_GetSvDiskType() 関数

機能説明

RAID カード (タイプ 2) に接続されているディスクのモデル名およびシリアル番号を取得します。

書式

```
#include "esm_api.h"
int ESM_GetSvDiskType(HANDLE hesm, int Array, int Drive, char *model, char *serial);
```

パラメーター

- HANDLE hesm : ESM_Open() で獲得したハンドルを指定します。
- int Array : 論理ボリュームの番号を指定します。0 ~ 3 の値が指定可能です。
- int Drive : ディスク番号を指定します。0 ~ 1 の値が指定可能です。
- char *model : モデル名を格納する領域へのポインターを指定します。最低 64byte (終端ヌル文字を含む) の領域を準備する必要があります。
- char *serial : シリアル番号を格納する領域へのポインターを指定します。最低 64byte (終端ヌル文字を含む) の領域を準備する必要があります。

復帰値

- ESM_SUCCESS(0) : 正常終了。
- ESM_ERROR(-1) : システムエラーが発生。
- ESM_PRMERR(-2) : 指定パラメーターに誤りがある。

留意事項

装置が RAID 構成でないまたは RAID 監視を無効に設定している場合、本関数は正常終了しますが、モデル名およびシリアル番号には空文字が格納されます。

3.18 ESM_GetSvDiskSmartAttr() 関数

機能説明

RAID カード（タイプ 2）に接続されているディスクの S.M.A.R.T. アトリビュートデータを取得します。

書式

```
#include "esm_api.h"
int ESM_GetSvDiskSmartAttr(HANDLE hesm, int Array, int Drive, smart_attribute_t *attributes);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
int Array	: 論理ボリュームの番号を指定します。0 ~ 3 の値が指定可能です。
int Drive	: ディスク番号を指定します。0 ~ 1 の値が指定可能です。
smart_attribute_t *attributes	: 情報を格納する領域へのポインター。 以下に示す smart_attribute_t 構造体へのポインターを指定します。 smart_attribute_t 配列 30 個分の格納領域を用意する必要があります。

```
smart_attribute_t 構造体
typedef struct smart_attribute {
    unsigned char id;
    unsigned char flags[2];
    unsigned char value;
    unsigned char worst;
    unsigned char threshold;
    unsigned char raw[7];
    unsigned char reserved[3];
} smart_attribute_t;
```

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。

留意事項

装置が RAID 構成でないまたは RAID 監視を無効に設定している場合、本関数は正常終了しますが、S.M.A.R.T. アトリビュートデータはすべて 0 の値が格納されます。

3.19 ESM_GetSwRAIDCount() 関数

機能説明

ソフトウェア RAID の、RAID を構成する物理ディスク数を取得します。例えば、RAID アレイが 1 つで、それを構成する物理ディスク数が 2 つの場合、2 を応答します。その場合、「ESM_GetSwRAIDInfo() 関数」、「ESM_GetSwDiskType() 関数」、「ESM_GetSwDiskSmartAttr() 関数」で指定する index は 0 または 1 となります。

呼び出し形式

```
#include "esm_api.h"  
int ESM_GetSwRAIDCount(HANDLE hesm, int *count);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
int *count	: ソフトウェア RAID の監視項目数を格納する領域へのポインターを指定します。

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。
ESM_NOSERVICE(-6)	: SysMon Entry のサービスが停止している。
ESM_NOINSTALL(-14)	: 装置にソフトウェア RAID がインストールされていない。
ESM_NOMONITOR(-15)	: ソフトウェア RAID 監視が無効になっている。

3.20 ESM_GetSwRAIDInfo() 関数

機能説明

ソフトウェア RAID の監視データを取得します。



index が 0 と 1 の物理ディスクによって RAID を構成している場合、0、1 どちらの index を指定しても「ESM_GetSwRAIDInfo() 関数」で得られる RAID に関する情報 (RAID の名前、RAID のリビルド進捗率、RAID のステータス、RAID レベル) は同一となります。

呼び出し形式

```
#include "esm_api.h"
int ESM_GetSwRAIDInfo(HANDLE hesm, unsigned int index, unsigned int *port_number,
                      esm_sw_raid_info_t *info);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
unsigned int index	: ソフトウェア RAID を構成する物理ディスクの識別番号を指定します。GUI 上の Index と同じです。
unsigned int *port_number	: ディスクのポート番号を格納する領域へのポインターを指定します。
esm_sw_raid_info_t *info	: 情報を格納する領域へのポインターを指定します。 以下に示す esm_sw_raid_info_t 構造体へのポインターを指定します。
esm_sw_raid_info_t 構造体 typedef struct { char array_name[16]; int disk_status; int smart_status; int array_status; int rebuild_progress; int raid_level; } esm_sw_raid_info_t;	

補足説明

esm_sw_raid_info_t 構造体の各メンバ変数の取りうる値について以下に示します。

変数名	意味	値	値に対応する GUI 表示	値の意味
array_name	RAID の名前	-	-	-

変数名	意味	値	値に対応する GUI表示	値の意味
disk_status	物理ディスクの ステータス	1	ONLINE	正常
		2	FAILED	故障
		3	REBUILDING	リビルド中
		4	NOARRAY	RAID 解体予約（データを引き継がないディスク）
		5	PRE INIT	RAID 解体予約（データを引き継ぐディスク）
smart_status	物理ディスクの S.M.A.R.T. ステータス	1	NORMAL	S.M.A.R.T. ステータスが正常
		2	ERROR	S.M.A.R.T. ステータスが異常
		3	N/A	S.M.A.R.T. ステータスが取得不可
array_status	RAID のステータス	1	OK	正常
		2	DEGRADED	RAID の冗長性が失われている
		3	REBUILDING	RAID がリビルド中
		4	FAILED	RAID が修復不能
		5	PRE DELETE	RAID の解体が予約されている
rebuild_progress	RAID の リビルド進捗率	-	-	-
raid_level	RAID レベル	1	RAID1	RAID1（ミラーリング）

復帰値

- ESM_SUCCESS(0) : 正常終了。
- ESM_ERROR(-1) : システムエラーが発生。
- ESM_PRMERR(-2) : 指定パラメーターに誤りがある。
- ESM_NOSERVICE(-6) : SysMon Entry のサービスが停止している。
- ESM_NOINSTALL(-14) : 装置にソフトウェア RAID がインストールされていない。
- ESM_NOMONITOR(-15) : ソフトウェア RAID 監視が無効になっている。
- ESM_NOT_FOUND(-16) : 指定した項目が存在しない。

3.21 ESM_GetSwDiskType() 関数

機能説明

ソフトウェア RAID のディスクのモデル名およびシリアル番号を取得します。

呼び出し形式

```
#include "esm_api.h"
int ESM_GetSwDiskType(HANDLE hesm, unsigned int index, unsigned int *port_number,
                      char *model, char *serial);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
unsigned int index	: ソフトウェア RAID を構成する物理ディスクの識別番号を指定します。GUI 上の Index と同じです。
unsigned int *port_number	: ディスクのポート番号を格納する領域へのポインターを指定します。
char *model	: モデル名を格納する領域へのポインターを指定します。最低 64byte (終端ヌル文字を含む) の領域を準備する必要があります。
char *serial	: シリアル番号を格納する領域へのポインターを指定します。最低 64byte (終端ヌル文字を含む) の領域を準備する必要があります。

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。
ESM_NOSERVICE(-6)	: SysMon Entry のサービスが停止している。
ESM_NOINSTALL(-14)	: 装置にソフトウェア RAID がインストールされていない。
ESM_NOMONITOR(-15)	: ソフトウェア RAID 監視が無効になっている。
ESM_NOT_FOUND(-16)	: 指定した項目が存在しない。

3.22 ESM_GetSwDiskSmartAttr() 関数

機能説明

ソフトウェア RAID のディスクの S.M.A.R.T. アトリビュートデータを取得します。

呼び出し形式

```
#include "esm_api.h"
int ESM_GetSwDiskSmartAttr(HANDLE hesm, unsigned int index, unsigned int *port_number,
                           smart_attribute_t *attributes);
```

パラメーター

HANDLE hesm	: ESM_Open() で獲得したハンドルを指定します。
unsigned int index	: ソフトウェア RAID を構成する物理ディスクの識別番号を指定します。GUI 上の Index と同じです。
unsigned int *port_number	: ディスクのポート番号を格納する領域へのポインターを指定します。
smart_attribute_t *attributes	: 情報を格納する領域へのポインター。 以下に示す smart_attribute_t 構造体へのポインターを指定します。 smart_attribute_t 配列 30 個分の格納領域を用意する必要があります。

smart_attribute_t 構造体

```
typedef struct smart_attribute {
    unsigned char id;
    unsigned char flags[2];
    unsigned char value;
    unsigned char worst;
    unsigned char threshold;
    unsigned char raw[7];
    unsigned char reserved[3];
} smart_attribute_t;
```

復帰値

ESM_SUCCESS(0)	: 正常終了。
ESM_ERROR(-1)	: システムエラーが発生。
ESM_PRMERR(-2)	: 指定パラメーターに誤りがある。
ESM_NOSERVICE(-6)	: SysMon Entry のサービスが停止している。
ESM_NOINSTALL(-14)	: 装置にソフトウェア RAID がインストールされていない。
ESM_NOMONITOR(-15)	: ソフトウェア RAID 監視が無効になっている。
ESM_NOT_FOUND(-16)	: 指定した項目が存在しない。

第4章 API 使用方法



この章では、EmbedWare/SysMon SDK の API 使用法について説明します。

4.1 API マルチスレッド対応

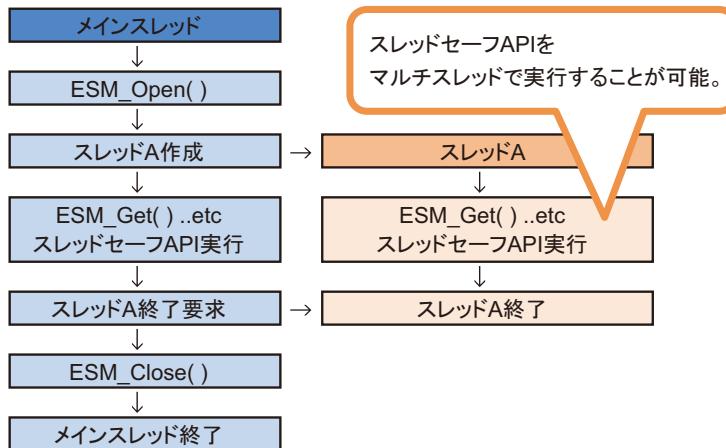
マルチスレッドでほかの API と同時に使用可能な API (以下、スレッドセーフ API) と、マルチスレッドでほかの API と同時に使用できない API (以下、スレッドアンセーフ API) が存在します。

スレッドセーフ API	スレッドアンセーフ API
ESM_Get() ESM_GetSmart() ESM_RAIDInfo() ESM_SvRAIDInfo() ESM_QueryError() ESM_GetWDTTimeOut() ESM_ReadGPI() ESM_ReadGPO() ESM_WriteGPO() ESM_GetCpuCount() ESM_GetCpuPerform() ESM_GetSmartEx() ESM_GetSvDiskType() ESM_GetSvDiskSmartAttr() ESM_GetSwRAIDCount() ESM_GetSwRAIDInfo() ESM_GetSwDiskType() ESM_GetSwDiskSmartAttr()	ESM_Open() ESM_Close() ESM_SwitchWDT() ESM_WDTHeartbeat()

4.2 API 使用例

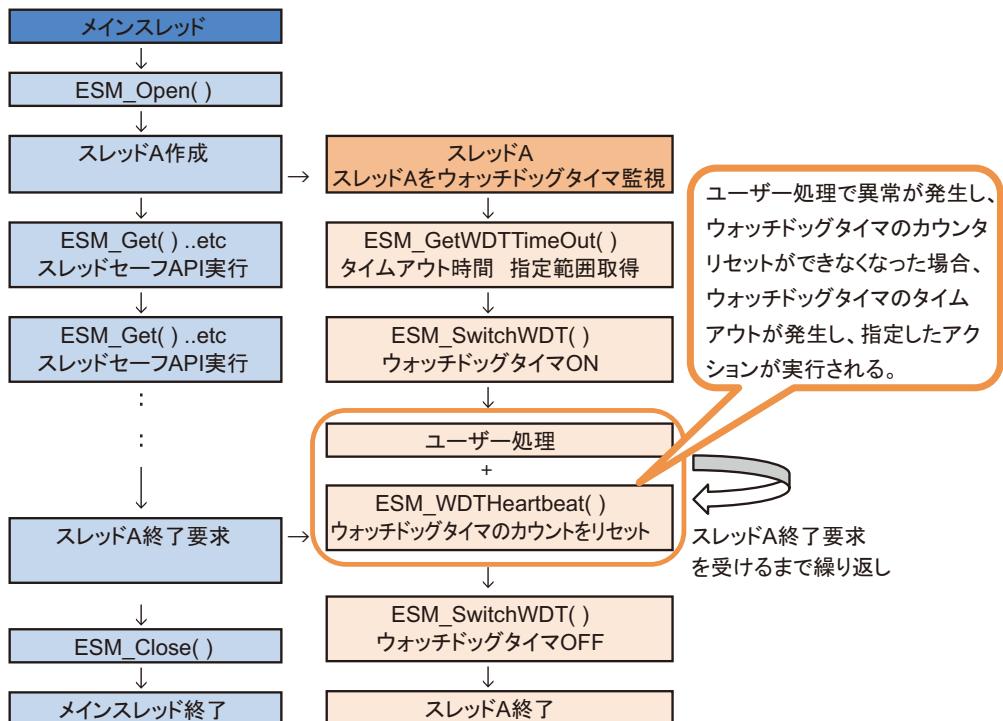
API 使用例を以下に示します。

◇ API 実行



◇ API を使用したウォッチドッグタイマ監視

スレッド A (ユーザー処理) のハングアップをウォッチドッグタイマで監視する例



付録 A 通知メッセージ一覧

EmbedWare/SysMon Entry で、コンピューターの動作状態が変更された場合に通知されるメッセージを、以下に示します。

◇情報

通知される情報メッセージを以下に示します。Windows イベントログ出力機能が有効な場合、Windows イベントログにメッセージが格納されます。

No.	メッセージ	メッセージ ID (*1)	意味
1	<センサー名> temperature is normal.	101	<センサー名> の温度が正常に戻った。
2	<センサー名> voltage is normal.	201	<センサー名> の電圧が正常に戻った。
3	<センサー名> fan speed is normal.	301	<センサー名> のファン回転数が正常に戻った。
4	SLOTX disk is normal.	401	HDD (非 RAID SATA ディスク) が復旧した。
5	PCI Error was reset.	501	PCI エラー検出状態をリセットした。
6	Memory Error was reset.	502	メモリエラー検出状態をリセットした。
7	BMC Error was reset.	503	BMC エラー検出状態をリセットした。
8	Chipset RAID: Information PORTX: <Status> : : RAID Card: Information CNTLXX CNXX DEVXX: <Status> : : SVRx RAID ArrayX: Information Drive0: <Model>, <SerialNumber>, <Status> Drive1: <Model>, <SerialNumber>, <Status>	601	チップセット RAID のディスク情報。装置起動時に出力される。 ディスク情報は監視デバイスの数分表示される。 RAID カード (タイプ 1) のディスク情報。装置起動時に出力される。ディスク情報は監視デバイスの数分表示される。 RAID カード (タイプ 2) のディスク情報。装置起動時に出力される。

No.	メッセージ	メッセージID (*1)	意味
9	RAID PORTX Rebuild start	602	チップセット RAID のディスクがリビルドを開始した。
	RAID CNTLXX CNXX DEVXX Rebuild start		RAID カード (タイプ 1) のディスクがリビルドを開始した。
	SVRx RAID ArrayX: Rebuild start Drive0: <Model>, <SerialNumber>, <Status> Drive1: <Model>, <SerialNumber>, <Status>		RAID カード (タイプ 2) のディスクがリビルドを開始した。
10	RAID PORTX Rebuild completed	603	チップセット RAID のディスクがリビルドを完了した。
	RAID CNTLXX CNXX DEVXX Rebuild completed		RAID カード (タイプ 1) のディスクがリビルドを完了した。
	SVRx RAID ArrayX: Rebuild completed Drive0: <Model>, <SerialNumber>, <Status> Drive1: <Model>, <SerialNumber>, <Status>		RAID カード (タイプ 2) のディスクがリビルドを完了した。
11	RAID PORTX is normal.	604	チップセット RAID のディスクが故障、または S.M.A.R.T. による異常から復旧。リビルド完了時も復旧したとみなす。
	RAID CNTLXX CNXX DEVXX is normal.		RAID カード (タイプ 1) のディスクが故障、または S.M.A.R.T. による異常から復旧。リビルド完了時も復旧したとみなす。
	SVRx RAID ArrayX: Normal Drive0: <Model>, <SerialNumber>, <Status> Drive1: <Model>, <SerialNumber>, <Status>		RAID カード (タイプ 2) のディスクが故障、または S.M.A.R.T. による異常から復旧。リビルド完了時も復旧したとみなす。
12	EWS Entry Service Start	901	EmbedWare/SysMon Entry のサービスが開始しました。
13	EWS Entry Service Stop	902	EmbedWare/SysMon Entry のサービスが停止しました。 シャットダウン時、Windows イベントログのサービスが先に終了する場合、情報出力されない。

*1 : メッセージ ID は、Windows イベントログに通知メッセージを格納する場合に有効です。



- "X" は可変の数値を表します。
- "<"、">" で囲んでいる箇所は可変の文字列を表します。

◇警告

通知される警告メッセージを以下に示します。Windows イベントログ出力機能が有効な場合、Windows イベントログにメッセージが格納されます。

No.	メッセージ	メッセージ ID (*1)	意味	対処
1	<センサー名> temperature upper threshold breached: XXX[degrees C]	1101	<センサー名>で温度異常を検出した。	吸気温度の場合は空調を、CPU 温度などの場合はファン状態を確認するか、または担当営業にご相談ください。
2	<センサー名> temperature is abnormal.	1102		
3	<センサー名> temperature lower threshold breached: XXX[degrees C]	1103		
4	<センサー名> voltage upper threshold breached: XXX[V]	1201	<センサー名>で電圧異常を検出した。	担当営業にご相談ください。
5	<センサー名> voltage lower threshold breached: XXX[V]	1202		
6	<センサー名> fan speed lower threshold breached: XXX[rpm]	1301	<センサー名>でファン回転数異常を検出した。	ファン故障の可能性があります。ファン回転数をご確認のうえ、担当営業にご相談ください。
7	<センサー名> fan speed upper threshold breached: XXX[rpm]	1302		
8	SLOTX disk is abnormal.	1401	S.M.A.R.T. 機能による故障予測を検出した。	ディスク交換を推奨します。詳細については、担当営業にご相談ください。
9	PCI Error [yy/mm/dd hh:mm:ss <詳細データ>]	1501	PCI エラーを検出した。	BIOS イベントログに PCI エラー発生記録が存在します。BIOS イベントログをご確認のうえ、担当営業にご相談ください。

No.	メッセージ	メッセージID (*1)	意味	対処
10	Memory Error [yy/mm/dd hh:mm:ss < 詳細データ >]	1502	メモリエラーを検出した。 メモリエラーが発生している場合、またはシングルビット ECC メモリエラーが頻発している場合はメモリ故障の可能性があるため、担当営業にご相談ください。	BIOS イベントログにメモリエラー発生記録が存在します。BIOS イベントログをご確認ください。マルチビット ECC メモリエラーが発生している場合、またはシングルビット ECC メモリエラーが頻発している場合はメモリ故障の可能性があるため、担当営業にご相談ください。
11	BMC Error [yy/mm/dd hh:mm:ss < 詳細データ >]	1503	BMC エラーを検出した。	BIOS イベントログに BMC エラー発生記録が存在します。BIOS イベントログをご確認のうえ、担当営業にご相談ください。
12	RAID PORTX is Failed.	1601	チップセット RAID のディスク故障を検出した。	ディスクを交換してください。
	RAID CNTLXX CNXX DEVXX is Failed.		RAID カード (タイプ 1) のディスク故障を検出した。	

No.	メッセージ	メッセージID (*1)	意味	対処
13	RAID PORTX is S.M.A.R.T. ERROR.	1602	チップセット RAID のディスクが S.M.A.R.T. による異常を検出した。	ディスク交換を推奨します。 詳細については、担当営業にご相談ください。
	RAID CNTLXX CNXX DEVXX is S.M.A.R.T. ERROR.		RAID カード (タイプ 1) のディスクが S.M.A.R.T. による異常を検出した。	
14 (*2)	SVRx RAID ArrayX: DriveX is Failed Drive0: <Model>, <SerialNumber>, <Status> Drive1: <Model>, <SerialNumber>, <Status>	1603	RAID カード (タイプ 2) のディスク故障を検出した。	ディスクを交換してください。
15 (*2)	SVRx RAID ArrayX: DriveX is S.M.A.R.T. ERROR Drive0: <Model>, <SerialNumber>, <Status> Drive1: <Model>, <SerialNumber>, <Status>	1604	RAID カード (タイプ 2) のディスクが S.M.A.R.T. による異常を検出した。	ディスク交換を推奨します。 詳細については、担当営業にご相談ください。
16 (*2)	SVRx RAID ArrayX: DriveX is SKIP RECON Drive0: <Model>, <SerialNumber>, <Status> Drive1: <Model>, <SerialNumber>, <Status>	1605	RAID カード (タイプ 2) のディスクがスキップ・リコンを検出した。	担当営業にご相談ください。
17 (*2)	Software RAID Array:<Name> PortX S.M.A.R.T. ERROR PortX: <Model>, <SerialNumber>, <Status> PortX: <Model>, <SerialNumber>, <Status>	1622	ソフトウェア RAID のディスクが S.M.A.R.T. による異常を検出した。	ディスク交換を推奨します。 詳細については、担当営業にご相談ください。

*1: メッセージ ID は、Windows イベントログに通知メッセージを格納する場合に有効です。

*2: EmbedWare/SysMon イベントログに出力される場合、改行コードは削除され、"["、"]" 文字で囲まれた形式で出力されます。Windows イベントログおよび表示メッセージボックスは、表に記載されているメッセージ形式で出力されます。



- "X" は可変の数値を表します。
- "<"、">" で囲んでいる箇所は可変の文字列を表します。
- "yy/mm/dd hh:mm:ss" は日時を表します。

◇エラー

通知される異常メッセージを以下に示します。Windows イベントログ出力機能の有効・無効にかかわらず、出力されます。

No.	メッセージ	メッセージ ID (*1)	意味	対処
1	BMC open error.	10901	BMC のオープンに失敗した。	IPMI ドライバが必須の OS の場合、IPMI ドライバが有効になっているかを確認してください。無効になっている場合は IPMI ドライバを有効にしてください。それでも問題が発生する場合は、担当営業にご相談ください。
2	Configuration file is broken.	10902	設定ファイルが壊れている。	担当営業にご相談ください。
3	Internal error.	10903	内部異常を検出した。	担当営業にご相談ください。

*1：メッセージ ID は、Windows イベントログに通知メッセージを格納する場合に有効です。

- Windows イベントログに関する補足

Windows イベントログにメッセージを格納する場合、イベント ID 以外のパラメーターは以下のとおりとなります。

種類	： 情報、警告、エラー
ソース	： EWS Entry Service
分類	： なし

付録 B GUI メッセージ一覧

EmbedWare/SysMon Entry の GUI 操作中に表示される可能性のあるメッセージを、以下に示します。

No.	メッセージ	意味	対処
1	記録をリセットしますか？	—	リセット実行するかどうかを選択してください。
2	BMC SEL をクリアしますか？	—	クリア実行するかどうかを選択してください。
3	ウォッチドッグタイマ設定は、コンピュータ再起動後に反映されます。	—	設定を反映するため、コンピューターを再起動してください。
4	RAID 設定は、コンピュータ再起動後に反映されます。	—	設定を反映するため、コンピューターを再起動してください。
5	<ファイル名> の保存に失敗しました。 <エラー内容>	ファイルの保存に失敗した。	エラー内容に応じて処置を行ってください。
6	EWS Entry Service が開始していません。 アプリケーションを終了します。	—	EmbedWare/SysMon Entry のサービスプログラムが初期処理中の可能性があります。しばらく待ってから、GUI 操作を実施してください。GUI 再操作で再びエラーが発生した場合は、担当営業にご相談ください。
7	しきい値が不正です。	入力された下限しきい値と上限しきい値の関係が不正。	正しいしきい値（下限しきい値 < 上限しきい値）を入力してください。
8	BMC アクセスエラー<エラーコード>。	—	実行中の BMC 処理終了後、再度実行してください。再実行でエラーが発生した場合は、担当営業にご相談ください。



"<"、">" で囲んでいる箇所は可変の文字列を表します。

参考

付録 C EmbedWare/SysMon SDK API ディスク情報取得について

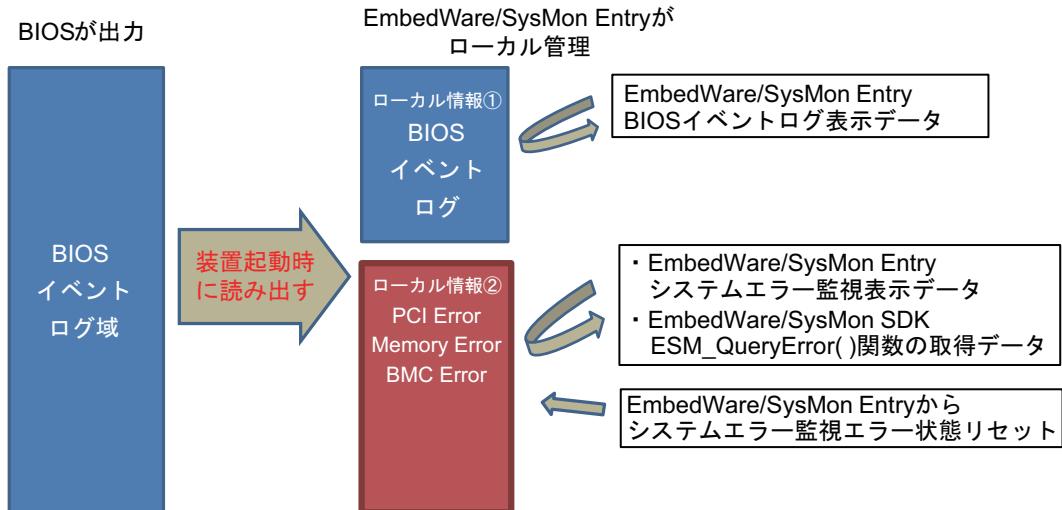
ディスク情報取得で使用する API を以下に示します。

取得する情報	モデル名・シリアル番号	S.M.A.R.T.ステータス	S.M.A.R.T.アトリビュートデータ
非 RAID ディスク	ESM_Get()	ESM_Get() ESM_GetSmart() ESM_GetSmartEx()	ESM_GetSmart() ESM_GetSmartEx()
RAID カード (タイプ1) ・チップセット RAID	取得不可	ESM_RAIDInfo()	取得不可
RAID カード (タイプ2)	ESM_GetSvDiskType()	ESM_SvRAIDInfo()	ESM_GetSvDiskSmartAttr()
ソフトウェア RAID	ESM_GetSwDiskType()	ESM_GetSwRAIDInfo()	ESM_GetSwDiskSmartAttr()

付録 D システムエラー監視の仕組みについて

EmbedWare/SysMon Entry のシステムエラー監視の仕組みを、以下に示します。

- EmbedWare/SysMon Entry は、装置起動時に BIOS イベントログ域からログを読み出し、ローカル情報①として保持します。この中でエラーを検出した場合、ローカル情報②にエラー情報を格納します。
- 以降の装置起動では、BIOS イベントログ域に新たに追加されたログだけを読み出し、ローカル情報①へ追加します。追加ログの中でエラーを検出した場合、ローカル情報②にエラー情報を格納します。
- BIOS イベントログ域をクリアしても、ローカル情報①②は削除しません。
- エラー検出時、すでに同種のシステムエラーがローカル情報②に格納済みの場合、新しく検出したエラーは格納しません。
- ローカル情報②に格納したシステムエラーは、EmbedWare/SysMon Entry からリセットできます。



EmbedWare/SysMon Ver 7.1 for Windows 説明書
P2WW-4170-02Z0

発行日 2019年11月

発行責任 株式会社 PFU

- 本書の内容は、改善のため事前連絡なしに変更することがあります。
- 本書に記載されたデータの使用に起因する、第三者の特許権およびその他の権利の侵害については、当社はその責を負いません。
- 無断転載を禁じます。