

次世代車載システム向け Watchdog Manager 外部仕様書

Ver.1.0.4

2017/03/06

Copyright (C) 2015-2017 by Center for Embedded Computing Systems

Graduate School of Information Science, Nagoya Univ., JAPAN

Copyright (C) 2015-2016 by AISIN COMCRUISE Co., Ltd., JAPAN

Copyright (C) 2015-2016 by eSOL Co., Ltd., JAPAN

Copyright (C) 2015-2017 by FUJI SOFT INCORPORATED, JAPAN

Copyright (C) 2015-2017 by NEC Communication Systems, Ltd., JAPAN

Copyright (C) 2015-2016 by Panasonic Advanced Technology Development Co., Ltd., JAPAN

Copyright (C) 2015-2017 by SCSK Corporation, JAPAN

Copyright (C) 2015-2016 by Sunny Giken Inc., JAPAN

Copyright (C) 2015-2017 by SUZUKI MOTOR CORPORATION

Copyright (C) 2015-2017 by TOSHIBA CORPORATION, JAPAN

Copyright (C) 2015-2017 by Witz Corporation

上記著作権者は、以下の (1)～(3)の条件を満たす場合に限り、本ドキュメント(本ドキュメントを改変したものを含む。以下同じ)を使用・複製・改変・再配布(以下、利用と呼ぶ)することを無償で許諾する。

- (1) 本ドキュメントを利用するには、上記の著作権表示、この利用条件および下記の無保証規定が、そのままの形でドキュメント中に含まれていること。
- (2) 本ドキュメントを改変する場合には、ドキュメントを改変した旨の記述を、改変後のドキュメント中に含めること。ただし、改変後のドキュメントが、TOPPERS プロジェクト指定の開発成果物である場合には、この限りではない。
- (3) 本ドキュメントの利用により直接的または間接的に生じるいかなる損害からも、上記著作権者および TOPPERS プロジェクトを免責すること。また、本ドキュメントのユーザまたはエンドユーザからのいかなる理由に基づく請求からも、上記著作権者および TOPPERS プロジェクトを免責すること。

本ドキュメントは、AUTOSAR(AUTomotive Open System ARchitecture)仕様に基いている。上記の許諾は、AUTOSAR の知的財産権を許諾するものではない。AUTOSAR は、AUTOSAR 仕様に基いたソフトウェアを商用目的で利用する者に対して、AUTOSAR パートナーになることを求めている。

本ドキュメントは、無保証で提供されているものである。上記著作権者および TOPPERS プロジェクトは、本ドキュメントに関して、特定の使用目的に対する適合性も含めて、いかなる保証も行わない。また、本ドキュメントの利用により直接的または間接的に生じたいかなる損害に関しても、その責任を負わない。

<目次>

1 概要	1
1.1 本文書の目的	1
1.2 関連文書	2
1.2.1 ベースとした文書	2
1.2.2 参考文書	2
1.3 凡例	3
1.3.1 仕様番号	3
1.3.2 API 仕様	4
1.3.3 注記	6
2 概念	7
2.1 機能一覧	7
2.2 一般的な挙動	8
2.3 ローカル監視状態	9
2.4 グローバル監視状態	14
2.5 監視機能	19
2.5.1 アライブ監視	19
2.5.1.1 アライブ監視のコンフィギュレーション	19
2.5.1.2 アライブ監視のアルゴリズム	20
2.5.2 デッドライン監視	21
2.5.2.1 デッドライン監視のコンフィギュレーション	21
2.5.2.2 デッドライン監視のアルゴリズム	22
2.5.3 ロジカル監視	24
2.5.3.1 ロジカル監視のコンフィギュレーション	24
2.5.3.2 ロジカル監視のアルゴリズム	25
2.6 エラーハンドリング / 復帰	28
2.6.1 RTE モードのメカニズム通知	28
2.6.2 WDM_GLOBAL_STATUS_STOPPED 時の DEM 報告	28
2.6.3 パーティションの再起動 / シャットダウン	28
2.6.4 トリガコンディション設定の停止	28
2.6.5 MCU リセット	29
2.7 Watchdog のトリガ操作	30
2.7.1 複数 Watchdog 対応	30
2.7.2 トリガコンディション設定	30
2.7.3 構成可能パラメータ	32
2.8 エラー仕様	32
2.8.1 エラー検出	32

2.8.2 エラー通知.....	3 2
2.9 モード切替.....	3 3
2.9.1 監視状態への影響	3 3
2.9.2 Watchdog への影響.....	3 4
2.9.3 スリープ中の Wacthdog 操作.....	3 5
2.10 ポートとポートインタフェースの仕様	3 5
2.10.1 アライブ監視のためのポートとポートインタフェース	3 6
2.10.1.1 一般的なアプローチ	3 6
2.10.1.2 データタイプ	3 7
2.10.1.3 アライブ監視のためのポートインタフェース	3 8
2.10.1.4 サービスポート	3 8
2.10.1.5 エラーコード	3 8
2.10.2 状態通知のためのポートとポートインタフェース	3 9
2.10.2.1 一般的なアプローチ	3 9
2.10.2.2 データタイプ	4 2
2.10.2.3 ポートインタフェース.....	4 2
2.10.2.4 モードポート	4 4
2.10.2.5 エラーコード	4 5
2.10.3 Watchdog Manager のサービスの定義	4 6
2.11 関連モジュール	4 8
2.11.1 Runtime Environment (RTE)	4 9
2.11.2 Watchdog Interface (WdgIf).....	4 9
2.11.3 ECU State Manager (EcuM)	4 9
2.11.4 BSW Scheduler (SchM)	4 9
2.11.5 OS.....	5 0
2.11.6 BSW Mode Manager (BswM)	5 0
2.11.7 Micro Controller Unit Driver (Mcu).....	5 0
2.11.8 Development Error Tracer (Det).....	5 0
2.11.9 Diagnostic Event Manager (Dem)	5 0
2.12 ファイル構成	5 1
2.12.1 ソースファイル構成	5 1
2.12.2 ヘッダーファイル構成.....	5 1
2.13 コンフィギュレーション.....	5 3
2.13.1 コンフィギュレーションクラス	5 3
2.13.2 コンフィギュレーションルール	5 3
2.13.3 公開パラメータ	5 3
2.13.4 モード非依存監視設定.....	5 4

2.13.4.1 監視エンティティ	5 4
2.13.4.2 OS アプリケーション.....	5 6
2.13.4.3 内部グラフのロジカル監視.....	5 7
2.13.5 モード依存パラメータ	5 8
2.13.5.1 モード	5 8
2.13.5.2 外部グラフのロジカル監視.....	5 9
2.13.5.3 アライブ監視	6 0
2.13.5.4 デッドライン監視	6 0
2.14 自己監視機能	6 1
2.14.1 WdgM_MainFunction コール周期/OS カウンタ監視機能.....	6 1
3 API 仕様	6 2
3.1 API 実装方針	6 2
3.1.1 実装言語	6 2
3.1.2 再入可能性.....	6 2
3.1.3 デバッグ	6 2
3.2 データ型	6 3
3.2.1 外部からインポートするデータ型.....	6 3
3.2.2 内部で定義するデータ型	6 4
3.2.2.1 WdgM_ConfigType	6 4
3.2.2.2 WdgM_SupervisedEntityIdType	6 4
3.2.2.3 WdgM_CheckpointIdType.....	6 4
3.2.2.4 WdgM_ModeType	6 5
3.2.2.5 WdgM_LocalStatusType	6 5
3.2.2.6 WdgM_GlobalStatusType	6 5
3.3 定数.....	6 6
3.3.1 WdgM_LocalStatusType	6 6
3.3.2 WdgM_GlobalStatusType	6 7
3.4 返り値	6 8
3.5 エラーコード	6 8
3.5.1 標準エラー.....	6 8
3.5.2 拡張エラー.....	6 9
3.6 API.....	7 0
3.6.1 WdgM_Init	7 0
3.6.2 WdgM_DeInit	7 2
3.6.3 WdgM_GetVersionInfo.....	7 4
3.6.4 WdgM_SetMode.....	7 5
3.6.5 WdgM_GetMode	7 7

3.6.6 WdgM_CheckpointReached	7 8
3.6.7 WdgM_GetLocalStatus	8 0
3.6.8 WdgM_GetGlobalStatus	8 2
3.6.9 WdgM_PerformReset	8 3
3.6.10 WdgM_GetFirstExpiredSEID	8 4
3.7 コールバック関数	8 5
3.8 周期関数	8 5
3.8.1 WdgM_MainFunction	8 5
3.9 依存インタフェース	8 7
3.9.1 必須インタフェース	8 7
3.9.2 任意インタフェース	8 9
3.9.3 コンフィギュレーション可能なインタフェース	8 9
3.10 コンテナおよびパラメータ	9 0
3.10.1 WdgM	9 0
3.10.2 WdgMGeneral	9 0
3.10.2.1 WdgMDefensiveBehavior	9 1
3.10.2.2 WdgMDemStoppedSupervisionReport	9 1
3.10.2.3 WdgMDevErrorDetect	9 2
3.10.2.4 WdgMImmediateReset	9 3
3.10.2.5 WdgMOffModeEnabled	9 3
3.10.2.6 WdgMDeadlineTimeoutEnabled	9 4
3.10.2.7 WdgMFCycleAndOsCounterSupervisionEnabled	9 4
3.10.2.8 WdgMVersionInfoApi	9 5
3.10.3 WdgMSupervisedEntity	9 5
3.10.3.1 WdgMSupervisedEntityId	9 6
3.10.3.2 WdgMEcuPartitionRef	9 6
3.10.3.3 WdgMInternalCheckpointInitialRef	9 7
3.10.3.4 WdgMInternalCheckpointFinalRef	9 7
3.10.3.5 WdgMOsApplicationRef	9 8
3.10.3.6 WdgMOSCounter	9 8
3.10.4 WdgMCheckpoint	9 9
3.10.4.1 WdgMCheckpointId	9 9
3.10.5 WdgMInternalTransition	1 0 0
3.10.5.1 WdgMInternalTransitionDestRef	1 0 0
3.10.5.2 WdgMInternalTransitionSourceRef	1 0 1
3.10.6 WdgMWatchdog	1 0 1
3.10.6.1 WdgMWatchdogName	1 0 2

3.10.6.2 WdgMWatchdogDeviceRef	1 0 2
3.10.7 WdgMConfigSet	1 0 3
3.10.7.1 WdgMInitialMode	1 0 3
3.10.8 WdgMDemEventParametersRefs	1 0 4
3.10.8.1 WDG_M_E_IMPROPER_CALLER	1 0 4
3.10.8.2 WDG_M_E_SUPERVISION	1 0 5
3.10.8.3 WDG_M_E_SET_MODE	1 0 5
3.10.8.4 WDG_M_E_OSCOUNTER	1 0 6
3.10.8.5 WDG_M_E_MFCYCLE_OR_OSCOUNTER	1 0 6
3.10.9 WdgMMode	1 0 7
3.10.9.1 WdgMExpiredSupervisionCycleTol	1 0 7
3.10.9.2 WdgMModelId	1 0 8
3.10.9.3 WdgMSupervisionCycle	1 0 8
3.10.10 WdgMAliveSupervision	1 0 9
3.10.10.1 WdgMExpectedAliveIndications	1 0 9
3.10.10.2 WdgMMaxMargin	1 1 0
3.10.10.3 WdgMMinMargin	1 1 0
3.10.10.4 WdgMSupervisionReferenceCycle	1 1 1
3.10.10.5 WdgMAliveSupervisionCheckpointRef	1 1 1
3.10.11 WdgMDeadlineSupervision	1 1 2
3.10.11.1 WdgMDeadlineMax	1 1 2
3.10.11.2 WdgMDeadlineMin	1 1 3
3.10.11.3 WdgMDeadlineStartRef	1 1 3
3.10.11.4 WdgMDeadlineStopRef	1 1 4
3.10.11.5 WdgMDeadlineTimeoutDetect	1 1 4
3.10.12 WdgMExternalLogicalSupervision	1 1 5
3.10.12.1 WdgMExternalCheckpointFinalRef	1 1 5
3.10.12.2 WdgMExternalCheckpointInitialRef	1 1 6
3.10.12.3 WdgMExternalTransition	1 1 6
3.10.12.3.1 WdgMExternalTransitionDestRef	1 1 7
3.10.12.3.2 WdgMExternalTransitionSourceRef	1 1 8
3.10.13 WdgMTrigger	1 1 9
3.10.13.1 WdgMTriggerConditionValue	1 1 9
3.10.13.2 WdgMWatchdogMode	1 2 0
3.10.13.3 WdgMTriggerWatchdogRef	1 2 0
3.10.14 WdgMMFCycleAndOsCounterSupervision	1 2 1
3.10.14.1 WdgMMFCycleMinTol	1 2 1

3.10.14.2 WdgMMFCycleMaxTol.....	1 2 1
3.10.14.3 WdgMOsCounterRef.....	1 2 2
3.10.15 WdgMLocalStatusParams.....	1 2 2
3.10.15.1 WdgMFailedAliveSupervisionRefCycleTol	1 2 2
3.10.15.2 WdgMLocalStatusSupervisedEntityRef.....	1 2 3
3.10.16 WdgMCallerIds.....	1 2 3
3.10.16.1 WdgMCallerId	1 2 3
4 リファレンス.....	1 2 4
4.1 データ型一覧	1 2 4
4.2 定数一覧	1 2 4
4.3 API 一覧.....	1 2 5
4.4 コールバック関数一覧.....	1 2 5
4.5 周期関数一覧	1 2 5
4.6 エラーコード一覧.....	1 2 6
4.7 仕様番号一覧	1 2 7
4.7.1 削除した AUTOSAR 仕様一覧.....	1 2 7
4.7.2 変更した AUTOSAR 仕様一覧.....	1 2 7
4.7.3 追加した NCES 仕様一覧.....	1 2 8



図 2.3-1：ローカル監視状態遷移	9
図 2.4-1：グローバル監視状態遷移	14
図 2.11-1：他モジュールとの関係	48
図 2.12-1：インクルード構成	52

表 1-1 ベースとした文書一覧.....	2
表 1-2 参考文書一覧	2
表 1-3 仕様番号凡例	3
表 2-1 ソースファイル	5 1
表 2-2 ヘッダーファイル.....	5 1
表 3-1 インポートするデータ型.....	6 3
表 3-2 Watchdog Manager システムサービスの返り値一覧	6 8
表 3-3 標準エラーコード一覧.....	6 8
表 3-4 拡張エラーコード一覧.....	6 9
表 3-5 必須インタフェース	8 7
表 3-6 任意インタフェース	8 9
表 4-1 データ型一覧	1 2 4
表 4-2 定数一覧	1 2 4
表 4-3 API 一覧.....	1 2 5
表 4-4 周期関数一覧	1 2 5
表 4-5 エラーコード一覧.....	1 2 6

1 概要

1.1 本文書の目的

本文書は Watchdog Manager の機能仕様を規定するものである。

本文書に含まれる仕様(以降、本仕様と略す)は、AUTOSAR Specification of Watchdog Manager 仕様(以降、AUTOSAR 仕様と略す)をベースに、必要な拡張と修正を行ったものである。

本文書に含まれる用語の定義は、「次世代車載システム向け Watchdog Manager 用語集」に記述している。

本文書は、Watchdog Manager に関する一般的な知識を持ったソフトウェア技術者が読むことを想定している。AUTOSAR 仕様に関する知識を持っていることが望ましいが、それを前提とせずに記述している。

1.2 関連文書

1.2.1 ベースとした文書

以下の表 1-1 は、本文書のベースとした文書であり、その内容は本文書内に含まれている。

表 1-1 ベースとした文書一覧

文書名	バージョン
AUTOSAR Specification of Watchdog Manager	V2.2.0 (R4.0 Rev3)

1.2.2 参考文書

以下の表 1-2 参考文書一覧表 1-2 は、本文書から参照している文書、または本文書を理解するために必要な文書である。その内容は本文書に含まれていない。

表 1-2 参考文書一覧

文書名	バージョン
次世代車載システム向け Watchdog Manager 用語集	Ver.1.0.0
次世代車載システム向け RTOS 外部仕様書	Ver3.2.0
次世代車載システム向け RTE 外部仕様書	Ver1.0.0
AUTOSAR Layered Architecture	V3.2.0 (R4.0 Rev3)
General Requirements on Basic Software Modules	V3.2.0 (R4.0 Rev3)
List of Basic Software Modules	V1.6.0 (R4.0 Rev3)
Software Component Template	V4.2.0 (R4.0 Rev3)
Generic Structure Template	V3.2.0 (R4.0 Rev3)
Requirements on System Template	V3.1.0 (R4.0 Rev3)
Requirements on Debugging	V1.0.0 (R4.0 Rev1)
Specification of BSW Module Description Template	V2.2.0 (R4.0 Rev3)
Specification of ECU Configuration	V3.2.0 (R4.0 Rev3)
Specification of Standard Types	V1.3.0 (R4.0 Rev1)
Requirements on Watchdog Driver	V2.1.0 (R4.0 Rev3)
Specification of Watchdog Manager	V2.2.0 (R4.0 Rev3)
Specification of Watchdog Interface	V2.5.0 (R4.0 Rev3)
Specification of Watchdog Driver	V2.5.0 (R4.0 Rev3)

1.3 凡例

1.3.1 仕様番号

本文書では、AUTOSAR 仕様と、名古屋大学大学院情報科学研究科附属組込みシステム研究センター (NCES) を中心とする車載制御システム向け高品質プラットフォームに関するコンソーシアム型共同研究 (AP (Automotive Platform) コンソーシアム) で新規に規定した仕様とが混在しているため、以下に示す仕様番号を用いてこれらの仕様を区別して管理を行う。仕様番号は、要求事項にのみ付与することを基本とする。ただし、AUTOSAR 仕様において、概念の説明や補足事項に対しても仕様番号が付与されているものに関しては、そのまま付与する。また、本文書から AP コンソーシアムで開発した A-WATCHDOGSTACK 情報として本文書に記載するため、仕様番号を付与する。

表 1-3 仕様番号凡例

仕様番号	内容
【WDGM _{xxx} 】	関連文書「AUTOSAR Specification of Watchdog Manager」で規定された仕様。AUTOSAR 仕様で記述されている Watchdog Manager 仕様番号を用いる。本仕様に採用しない AUTOSAR 仕様についても【WDGM _{xxx} 】で記載している。
【WDGM_Conf _{xxx} 】	関連文書「AUTOSAR Specification of Watchdog Manager」で規定された仕様。AUTOSAR 仕様に記述されているコンフィギュレーション仕様番号を用いる。本仕様に採用しない AUTOSAR 仕様についても【WDGM_Conf _{xxx} 】で記載している。
【WDGM _a _{xxx} 】	関連文書「AUTOSAR Specification of Watchdog Manager」で規定された機能仕様のうち、仕様番号が付与されていないもの。AUTOSAR 仕様の要求事項であるため、本仕様において仕様番号を付与する。
【NWDGM _{xxx} 】	AP コンソーシアムで新規に規定した仕様。
【NWDGM_Conf _{xxx} 】	AP コンソーシアムで新規に規定したコンフィギュレーション仕様。
[WDGM _{xxx}] [NWDGM _{xxx}]	本文書で上記の仕様番号を内部参照する際に使用する。 仕様定義である【WDGM _{xxx} 】【NWDGM _{xxx} 】と区別するために左記のように記述する。
<WDGM _{xxx} >	本仕様で削除した AUTOSAR 仕様の仕様番号を記載する際に使用する。

1.3.2 API 仕様

データ型名

データ型名	＜データ型の ID 名を記載する＞
型	＜データ型の種類を記載する＞
範囲	＜そのデータ型の値の範囲を記載する＞
概要	＜データ型が示す内容を記載する＞

定数名

定数名	＜定数の ID 名を記載する＞
概要	＜定数が示す内容を記載する＞

システムサービス名

サービス ID	＜システムサービスの ID を記載する＞
C 言語 I/F	＜システムサービスの C 言語 I/F を記載する＞
パラメータ	[in] ＜システムサービスのパラメータで入力となるものを記載する＞
	[in/out] ＜システムサービスのパラメータで入力と出力となるものを記載する＞
	[out] ＜システムサービスのパラメータで出力となるものを記載する＞
返り値	＜システムサービスの返り値で標準エラーとなるものと、該当エラーが発生する条件を記載する。(W)が付いている場合は警告として扱っても良い＞
DET エラー	＜システムサービス内で通知する DET エラーのエラーコードと発生条件を記載する＞
同期/非同期	＜システムサービスの同期性を記載する＞
再入可能/再入不可能	＜システムサービスの再入可能性を記載する＞
機能	＜システムサービスの機能を記載する＞

メイン関数名

サービス ID	＜メイン関数の ID を記載する＞
C 言語 I/F	＜メイン関数の C 言語 I/F を記載する＞
起動タイミング	＜メイン関数の起動タイミングを記載する＞
DET エラー	＜システムサービス内で通知する DET エラーのエラーコードと発生条件を記載する＞
機能	＜システムサービスの機能を記載する＞

コンテナ名<コンテナ名称の末尾のみを記載する>

コンテナ名	<コンテナ名称をフルパスで記載する>
概要	<コンテナの概要を記載する>
多重度	<コンテナの多重度を記載する>
パラメータ	<コンテナが含むパラメータ名を列挙する>
サブコンテナ	<コンテナが含むサブコンテナ名を列挙する>

パラメータ<パラメータ名称の末尾のみを記載する>

パラメータ名	<パラメータ名称をフルパスで記載する>
概要	<パラメータの概要を記載する>
多重度	<パラメータの多重度を記載する>
型	<パラメータの型を記載する>
値の範囲	<パラメータの値の範囲を記載する>
デフォルト値	<パラメータのデフォルト値を記載する>
コンフィギュレーションクラス	
<プレコンパイルタイム，リンクタイム，ポストビルドタイムのいずれかを記載する>	
制限事項	
<パラメータの制限事項を記載する>	

1.3.3 注記

AUTOSAR 仕様との違い

削除や改変を行った AUTOSAR 仕様に対して，どのような差分があるかを説明する．

設計上の注意

本仕様に準拠した Watchdog Manager(以降, 本 Watchdog Manager と略す)を開発するユーザに対する注意事項または推奨事項を説明する．本 Watchdog Manager の使用者に向けた項目ではない．

使用上の注意

本 Watchdog Manager を使用してアプリケーションを開発するユーザに対する注意事項または推奨事項を説明する．本 Watchdog Manager の開発者に向けた項目ではない．

2 概念

2.1 機能一覧

本 Watchdog Manager が提供する機能の概略を以下に示す。

監視機能

プログラム実行を監視して時間的/論理的な制約違反を検出する。

Watchdog 操作

Watchdog Interface を介して，Watchdog Driver にトリガコンディションを報告する。

エラーハンドリング

監視結果に基づき，エラー通知や監視異常からの復帰処理をする。

2.2 一般的な挙動

Watchdog Manager が監視する単位を監視エンティティと呼ぶ。各監視エンティティは異なる監視機能、およびその組み合わせで監視することができる。

本 Watchdog Manager が有する監視機能は以下の通りである。

- ・アライブ監視(2.5.1 参照)
- ・デッドライン監視(2.5.2 参照)
- ・ロジカル監視(2.5.3 参照)

各監視機能は各監視エンティティの監視結果(正常/異常)を生成する。すなわち、すべての監視エンティティにアライブ監視結果、デッドライン監視結果、ロジカル監視結果が存在する。

また、Watchdog Manager の初期化時にすべての監視結果を正常にセットする。

与えられたモード内で、各監視エンティティは 0～複数個のアライブ監視(WdgMAliveSupervision)を持つこともあり、それぞれが監視結果(正常/異常)を持つ。

与えられたモード内で、それぞれの監視エンティティは 0～複数個のデッドライン監視(WdgMDeadlineSupervision)を持つこともあり、それぞれが監視結果(正常/異常)を持つ。

ロジカル監視には外部ロジカル監視と内部ロジカル監視の 2 種類がある。

与えられたモード内で、各監視エンティティは 0～複数個のロジカル監視(WdgMExternalLogicalSupervision/ WdgMInternalTransition)を持つこともあり、それぞれが監視結果(正常/異常)を持つ。

監視機能の監視結果に基づいて、各監視エンティティのローカル監視状態を決定する。

同一 MCU 内に所属する監視エンティティのローカル監視状態に基づいて、グローバル監視状態を決定する。

2.3 ローカル監視状態

ローカル監視状態は、以下の情報に基づいて決定する

1. ローカル監視状態の前回値
2. 各監視機能(アライブ監視, デッドライン監視, ロジカル監視)の監視結果の現在値

ローカル監視状態の遷移は WdgM_MainFunction 関数によって行われる。

ローカル監視状態の初期化は WdgM_Init 関数で行われる。

アライブ監視では、パラメータ WdgMFailedAliveSupervisionRefCycleTol に失敗を許容される回数が構成されており、失敗回数がそれ以下の時は状態 WDGM_LOCAL_STATUS_FAILED となる。

Watchdog Manager は各監視エンティティのローカル監視状態を追跡するものとする。ローカル監視状態は WdgM_LocalStatusType(3.3.1 参照)により取得可能である【WDGM200】。

各監視エンティティのローカル監視状態は Watchdog Manager のデバッグのために利用可能とする(3.1.3 参照)。下図にローカル監視状態の状態遷移図を示す【WDGM238】。

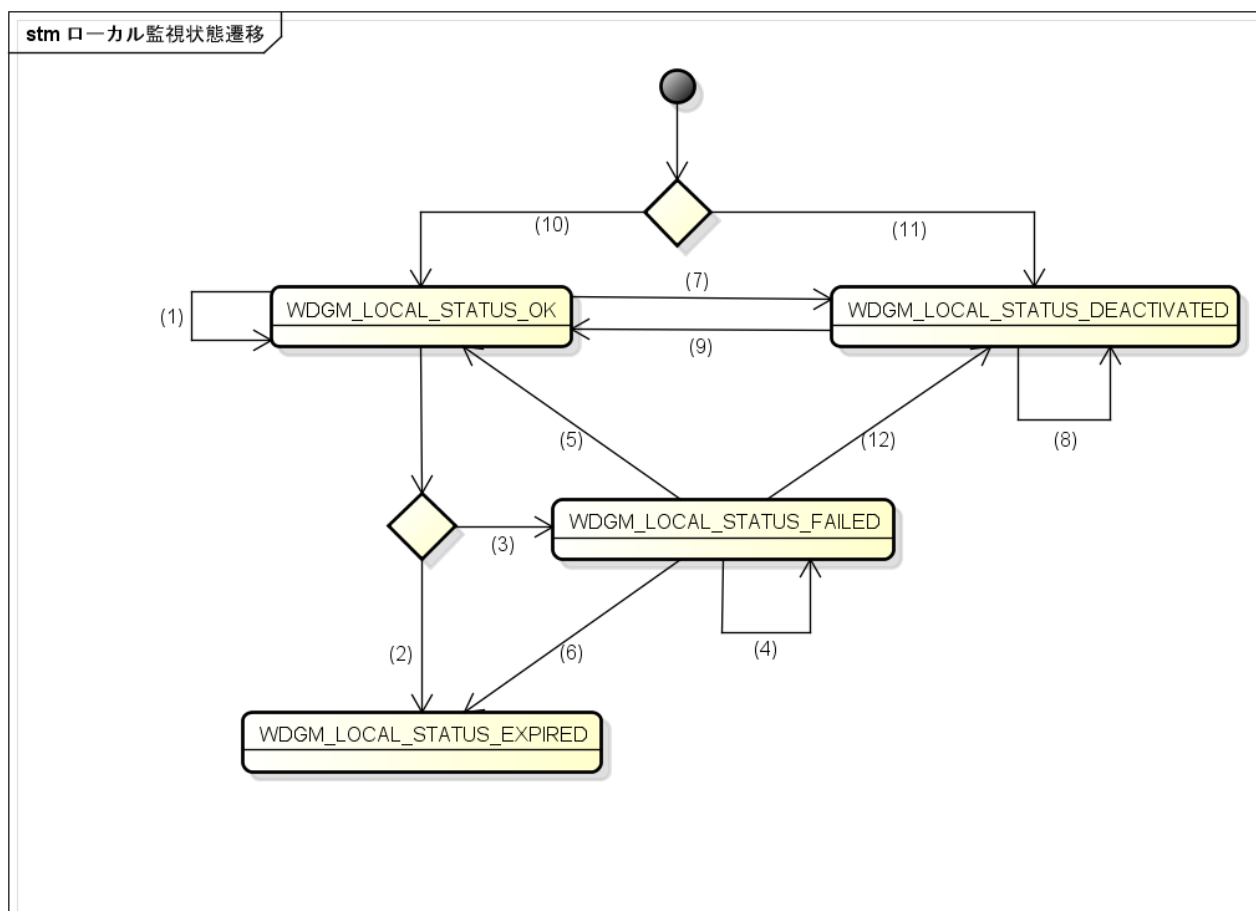


図 2.3-1 : ローカル監視状態遷移

以下にローカル監視状態の遷移条件について記載する。

(1) Wdgm_Local_Status_OK から Wdgm_Local_Status_OK への遷移【WDGM201】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：全ての監視機能の監視結果が正常である場合

遷移時アクション：なし

(2) Wdgm_Local_Status_OK から Wdgm_Local_Status_Expired への遷移【WDGM202】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：以下のいずれかの条件を満たす場合

- ・少なくとも1つのアライブ監視の監視結果が異常であり、
許容回数(WdgMFailedAliveSupervisionRefCycleTol)が0である
- ・少なくとも1つのデッドライン監視の監視結果が異常である
- ・少なくとも1つのロジカル監視の監視結果が異常である

遷移時アクション：なし

(3) Wdgm_Local_Status_OK から Wdgm_Local_Status_Failed への遷移【WDGM203】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：以下の全ての条件を満たす場合

- ・少なくとも1つのアライブ監視の監視結果が異常であり、
許容回数(WdgMFailedAliveSupervisionRefCycleTol)が0より大きい
- ・全てのデッドライン監視の監視結果が正常である
- ・全てのロジカル監視の監視結果が正常である

遷移時アクション：アライブ監視異常検出カウンタインクリメント

(4) WDGM_LOCAL_STATUS_FAILED から WDGM_LOCAL_STATUS_FAILED への遷移

条件 1：アライブ監視の監視結果が異常の場合【WDGM204】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：以下の全ての条件を満たす場合

- ・少なくとも 1 つのアライブ監視の監視結果が異常であり、
アライブ監視異常検出カウンタの値が許容回数
(WdgMFailedAliveSupervisionRefCycleTol)を上回っていない
- ・全てのデッドライン監視の監視結果が正常である
- ・全てのロジカル監視の監視結果が正常である

遷移時アクション：アライブ監視異常検出カウンタインクリメント

条件 2：アライブ監視の監視結果が正常の場合【WDGM300】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：以下の全ての条件を満たす場合

- ・全てのアライブ監視の監視結果が正常であり、
アライブ監視異常検出カウンタの値が 1 より大きい
- ・全てのデッドライン監視の監視結果が正常である
- ・全てのロジカル監視の監視結果が正常である

遷移時アクション：アライブ監視異常検出カウンタデクリメント

(5) WDGM_LOCAL_STATUS_FAILED から WDGM_LOCAL_STATUS_OK への遷移【WDGM205】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：以下の全ての条件を満たす場合

- ・全てのアライブ監視の監視結果が正常であり、
アライブ監視異常検出カウンタの値が 1 である
- ・全てのデッドライン監視の監視結果が正常である
- ・全てのロジカル監視の監視結果が正常である

遷移時アクション：アライブ監視異常検出カウンタデクリメント

(6) WDGM_LOCAL_STATUS_FAILED から WDGM_LOCAL_STATUS_EXPIRED への遷移

【WDGM206】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：以下のいずれかの条件を満たす場合

- ・少なくとも 1 つのアライブ監視の監視結果が異常であり、
アライブ監視異常検出カウンタの値が許容回数
(WdgMFailedAliveSupervisionRefCycleTol)を上回っている
- ・少なくとも 1 つのデッドライン監視の監視結果が異常である
- ・少なくとも 1 つのロジカル監視の監視結果が異常である

遷移時アクション：なし

(7) WDGM_LOCAL_STATUS_OK から WDGM_LOCAL_STATUS_DEACTIVATED への遷移

【WDGM207】

遷移が実行されるシステムサービス：WdgM_SetMode

遷移条件：非アクティブに切り替えられた場合

遷移時アクション：なし

(8) WDGM_LOCAL_STATUS_DEACTIVATED から WDGM_LOCAL_STATUS_DEACTIVATED への遷移 【WDGM208】

遷移が実行されるシステムサービス：WdgM_MainFunction, WdgM_CheckpointReached

遷移条件：必ず

遷移時アクション：監視を実施しない

(9) WDGM_LOCAL_STATUS_DEACTIVATED から WDGM_LOCAL_STATUS_OK への遷移

【WDGM209】

遷移が実行されるシステムサービス：WdgM_SetMode

遷移条件：アクティブに切り替えられた場合

遷移時アクション：なし

(10)初期状態から WDGM_LOCAL_STATUS_OK への遷移【WDGM268】

遷移が実行されるシステムサービス：WdgM_Init

遷移条件：初期モード(WdgMInitialMode)で各エンティティが参照されている場合
(初期モードで各監視エンティティがアクティブな場合)

遷移時アクション：なし

(11)初期状態から WDGM_LOCAL_STATUS_DEACTIVATED への遷移【WDGM269】

遷移が実行されるシステムサービス：WdgM_Init

遷移条件：初期モード(WdgMInitialMode)で各エンティティが参照されていない場合
(初期モードで各監視エンティティが非アクティブな場合)
(10)の遷移条件を満たさない場合

遷移時アクション：なし

(12)WDGM_LOCAL_STATUS_FAILED から WDGM_LOCAL_STATUS_DEACTIVATED への遷移
【WDGM291】

遷移が実行されるシステムサービス：WdgM_SetMode

遷移条件：非アクティブに切り替えられた場合

遷移時アクション：なし

2.4 グローバル監視状態

グローバル監視状態は所属する MCU 内の全ての監視エンティティのローカル監視状態を基に計算される。

下図にグローバル監視状態の状態遷移図を示す。

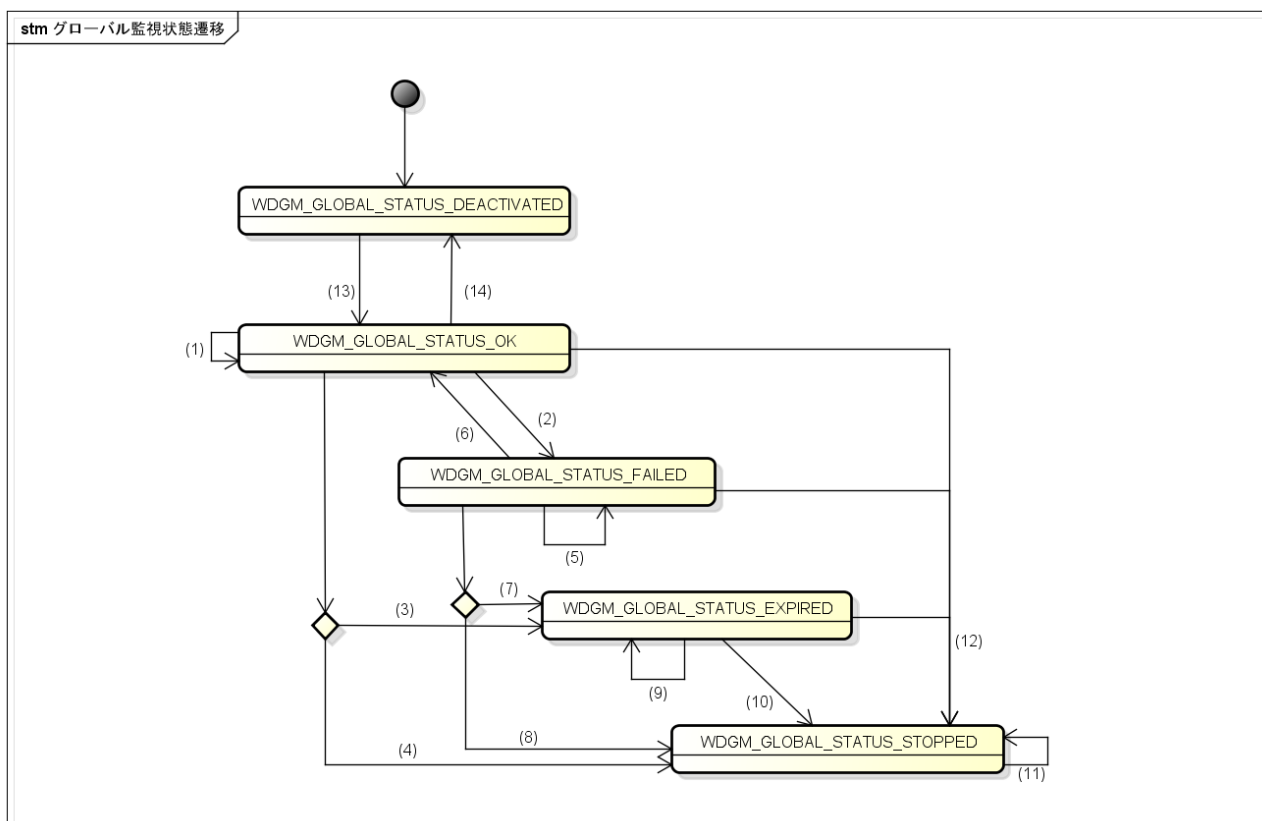


図 2.4-1：グローバル監視状態遷移

Watchdog Manager は監視するソフトウェア全体に対して、グローバル監視状態を 1 つ持つものとする【WDGM213】。

グローバル監視状態は Watchdog Manager のデバッグのために利用可能とする(3.1.3 参照)

【WDGM239_1】。

Watchdog Manager は WdgMExpiredSupervisionCycleTol で構成された許容回数の分だけ、エラー対応を遅延させる機能を提供する。遅延させることができる時間は監視サイクルの倍数であり、WdgM_MainFunction 関数がコールされるごとにカウントされる。許容回数はグローバル監視状態のステートマシン内に実装されている。遅延されている間のグローバル監視状態は WDGM_GLOBAL_STATUS_EXPIRED である【WDGM239_2】。

WdgM_MainFunction 関数は監視サイクルごとにグローバル監視状態の計算を行うものとする。WdgM_MainFunction 関数はローカル監視状態の計算を行ったあと、グローバル監視状態の計算を行う。グローバル監視状態の更新周期は WDGM_GLOBAL_STATUS_EXPIRED から WDGM_GLOBAL_STATUS_STOPPED への遷移をタイマーにトリガする必要がある【WDGM214】。

以下にグローバル監視状態の遷移条件を記載する。

- (1) WDGM_GLOBAL_STATUS_OK から WDGM_GLOBAL_STATUS_OK への遷移【WDGM078】
遷移が実行されるシステムサービス：WdgM_MainFunction
遷移条件：全ての監視エンティティのローカル監視状態が WDGM_LOCAL_STATUS_OK, または WDGM_LOCAL_STATUS_DEACTIVATED である場合
遷移時アクション：なし
- (2) WDGM_GLOBAL_STATUS_OK から WDGM_GLOBAL_STATUS_FAILED への遷移【WDGM076】
遷移が実行されるシステムサービス：WdgM_MainFunction
遷移条件：以下の全ての条件を満たす場合
 - ・少なくとも 1 つの監視エンティティのローカル監視状態が WDGM_LOCAL_STATUS_FAILED である
 - ・全ての監視エンティティのローカル監視状態が WDGM_LOCAL_STATUS_EXPIRED でない遷移時アクション：なし

メモ：

Watchdog Manager はエラー対応を遅延される機能をサポートする。これは watchdog リセットの前にエラー原因や NVRAM データの書き込みなどの、クリーンアップ動作を行うのに使用できる。

(3) WDM_GLOBAL_STATUS_OK から WDM_GLOBAL_STATUS_EXPIRED への遷移

【NWDGM105】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：少なくとも 1 つの監視エンティティのローカル監視状態が

WDM_LOCAL_STATUS_EXPIRED であり，許容回数

(WdgMExpiredSupervisionCycleTol)が 0 より大きな値で構成されている場合

遷移時アクション：Expired カウンタをインクリメントする

(4) WDM_GLOBAL_STATUS_OK から WDM_GLOBAL_STATUS_STOPPED への遷移

【WDGM216】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：少なくとも 1 つの監視エンティティのローカル監視状態が

WDM_LOCAL_STATUS_EXPIRED であり，許容回数

(WdgMExpiredSupervisionCycleTol)が 0 で構成されている場合

遷移時アクション：なし

(5) WDM_GLOBAL_STATUS_FAILED から WDM_GLOBAL_STATUS_FAILED への遷移

【WDGM217】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：以下の全ての条件を満たす場合

- ・少なくとも 1 つの監視エンティティのローカル監視状態が

WDM_LOCAL_STATUS_FAILED である

- ・全ての監視エンティティのローカル監視状態が WDM_LOCAL_STATUS_EXPIRED でない

遷移時アクション：なし

(6) WDM_GLOBAL_STATUS_FAILED から WDM_GLOBAL_STATUS_OK への遷移

【WDGM218】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：全ての監視エンティティのローカル監視状態が WDM_LOCAL_STATUS_OK，または

WDM_LOCAL_STATUS_DEACTIVATED である場合

遷移時アクション：なし

(7) WDGM_GLOBAL_STATUS_FAILED から WDGM_GLOBAL_STATUS_EXPIRED への遷移
【NWDGM048】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：少なくとも 1 つの監視エンティティのローカル監視状態が

WDGM_LOCAL_STATUS_EXPIRED であり，許容回数

(WdgMExpiredSupervisionCycleTol)が 0 より大きな値で構成されている場合

遷移時アクション：Expired カウンタをインクリメントする

(8) WDGM_GLOBAL_STATUS_FAILED から WDGM_GLOBAL_STATUS_STOPPED への遷移
【WDGM117】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：少なくとも 1 つの監視エンティティのローカル監視状態が

WDGM_LOCAL_STATUS_EXPIRED であり，許容回数

(WdgMExpiredSupervisionCycleTol)が 0 で構成されている場合

遷移時アクション：なし

(9) WDGM_GLOBAL_STATUS_EXPIRED から WDGM_GLOBAL_STATUS_EXPIRED への遷移
【NWDGM049】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：Expired カウンタの値が許容回数(WdgMExpiredSupervisionCycleTol)未満である場合

遷移時アクション：Expired カウンタをインクリメントする

(10) WDGM_GLOBAL_STATUS_EXPIRED から WDGM_GLOBAL_STATUS_STOPPED への遷移
【NWDGM050】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：Expired カウンタの値が許容回数(WdgMExpiredSupervisionCycleTol)以上である場合

遷移時アクション：なし

(11) WDGM_GLOBAL_STATUS_STOPPED から WDGM_GLOBAL_STATUS_STOPPED への遷移
【WDGM221】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：必ず

遷移時アクション：なし

(12)WdgM_SetMode 失敗時の遷移【WDGM139】

遷移が実行されるシステムサービス：WdgM_SetMode

遷移条件：WdgIf_SetMode が失敗した場合

遷移時アクション：なし

WdgM_MainFunction コール周期/OS カウンタ監視機能異常検出時の遷移【NWDGM084】

遷移が実行されるシステムサービス：WdgM_MainFunction

遷移条件：WdgM_MainFunction コール周期/OS カウンタ監視機能異常検出したとき

遷移時アクション：なし

OS カウンタのサービス(GetCounterValue, GetElapsedValue)失敗時の遷移 [NWDGM093]

遷移が実行されるシステムサービス：WdgM_CheckpointReached, WdgM_MainFunction

遷移条件：OS カウンタのサービスの返り値が E_OK でなかった場合

遷移時アクション：なし

(13)WDGM_GLOBAL_STATUS_DEACTIVATED から WDGM_GLOBAL_STATUS_OK への遷移
【WDGM285】

遷移が実行されるシステムサービス：WdgM_Init

遷移条件：WdgM_Init が正常にコールされた場合

遷移時アクション：なし

(14)WDGM_GLOBAL_STATUS_OK から WDGM_GLOBAL_STATUS_DEACTIVATED への遷移
【WDGM286】

遷移が実行されるシステムサービス：WdgM_DeInit

遷移条件：WdgM_DeInit が正常にコールされた場合

遷移時アクション：なし

2.5 監視機能

2.5.1 アライブ監視

アライブ監視は、独立した監視エンティティの周期的なタイミング制約のチェックを行う監視機能である。

2.5.1.1 アライブ監視のコンフィギュレーション

アライブ監視には以下の設定を与えることができる。

- WdgMExpectedAliveIndications [WDGM_Conf 311]
アライブ監視サイクル内のチェックポイントの到達回数
- WdgMMaxMargin [WDGM_Conf 313]
アライブ監視サイクル内のチェックポイントの到達回数のマージンの上限
- WdgMMinMargin [WDGM_Conf 312]
アライブ監視サイクル内のチェックポイントの到達回数のマージンの下限
- WdgMSupervisionReferenceCycle [WDGM_Conf 310]
アライブ監視サイクル(アライブ監視一周期内で WdgM_MainFunction がコールされる回数)

なお、上記の設定は全て、独立したチェックポイントに対してそれぞれ与えることができる。

Watchdog Manager はアライブ監視の実施に必要な独立したデータリソースの量を計算する。データリソースの量は監視エンティティ数、WdgMModes の数、アライブ監視の数等から計算する。Watchdog Manager のデータリソースの例としては、アライブカウンタ、FAILED カウンタ、EXPIRED カウンタ等がある【WDGM085】。

各チェックポイントのアライブカウンタは Watchdog Manager のデバッグに利用可能とする【WDGM240】。

2.5.1.2 アライブ監視のアルゴリズム

監視エンティティがアライブ通知を送信するには、WdgM_CheckpointReached 関数をコールする。それを受けて Watchdog Manager は、そのチェックポイントのアライブカウンタをインクリメントする。

WdgM_MainFunction 関数は WdgMSupervisionCycle で設定された監視周期で AUTOSAR Scheduler により周期的に実行される。

WdgM_MainFunction によるアライブカウンタの検証は WdgMSupervisionReferenceCycle で設定される周期で行う。

WdgM_MainFunction 関数はアクティブなモード内で構成されている各アライブ監視を実施するものとし、監視エンティティの各チェックポイントのアライブカウンタの検証を行う。その検証は該当するアライブ監視の WdgMSupervisionReferenceCycle で設定される周期で行う。アライブ監視の監視周期の間では WdgM_MainFunction 関数はアライブカウンタの検証を行わない【WDGM098】。

WdgM_MainFunction 関数ではアライブカウンタが以下に示す許容回数の範囲内に収まっているかを検証する。アライブカウンタが範囲内に収まっていなかった場合、その監視エンティティを異常と判定する【WDGM074】。

アライブカウンタの許容回数の範囲：

上限：WdgMExpectedAliveIndications + WdgMMaxMargin

下限：WdgMExpectedAliveIndications - WdgMMinMargin

WdgM_MainFunction はアライブカウンタが許容回数の範囲内であれば正常、そうでなければ異常と判定する。アクティブな Watchdog Manager モードでアライブ監視が構成されていないチェックポイントは無視する【WDGM115】。

アクティブな Watchdog Manager モードで対応するアライブ監視が定義されていない場合、WdgM_MainFunction はアライブカウンタの検証を行わないものとする【WDGM083】。

2.5.2 デッドライン監視

デッドライン監視は周期的でない監視エンティティのタイミング制約をチェックする。これらの監視エンティティでは与えられたタイムスパンの範囲内で、あるイベントが発生してからそれに追従するイベントが発生する。このタイムスパンは下限～上限の幅で与えることができる。

2.5.2.1 デッドライン監視のコンフィギュレーション

デッドライン監視には以下の設定を与えることができる。

- WdgMDeadlineStartRef [WDGM_Conf315]
デッドライン監視開始チェックポイントへの参照
- WdgMDeadlineStopRef [WDGM_Conf316]
デッドライン監視終了チェックポイントへの参照
- WdgMDeadlineMin [WDGM_Conf317]
デッドライン始端
- WdgMDeadlineMax [WDGM_Conf318]
デッドライン終端
- WdgMDeadlineTimeoutDetect [NWDGM_Conf075]
WdgM_MainFunction によるタイムアウト検出の有無

Watchdog Manager はデッドライン監視の実施に必要な独立したデータリソースの量を計算する。データリソースの量は監視エンティティ数、WdgMModes の数、デッドライン監視の数等から計算する【WDGM293_1】。

2.5.2.2 デッドライン監視のアルゴリズム

WdgM_Init 関数は全てのデッドライン監視開始チェックポイントのタイムスタンプを 0 に初期化する。デッドライン監視開始チェックポイントに到達したとき、監視エンティティが WdgM_CheckpointReached 関数をコールし、Watchdog Manager はデッドライン監視を実行する【WDGM298】。

WdgM_Init 関数は全てのデッドライン監視開始チェックポイント到達フラグを FALSE に初期化する【NWDGM053】。

これは、タイムスタンプが 0 かどうかによって開始チェックポイントが到達しているかどうかを判断すると、開始チェックポイント到達時に取得したタイムスタンプが 0 だった場合、終了チェックポイント到達時に監視を実行できないためである。

デッドライン監視で取得するタイムスタンプは OS カウンタのサービス(GetCounterValue, GetElapsedValue)によって取得する【NWDGM088】。

デッドライン監視開始チェックポイントが到達し、そのチェックポイントがアクティブなモードで参照されているとき、WdgM_CheckpointReached 関数はそのデッドライン監視開始チェックポイントの元に現在のタイムスタンプを記憶するものとする。ここで記憶したタイムスタンプは、対応するデッドライン監視終了チェックポイント到達時に経過時間を検証するために参照する【WDGM228】。

デッドライン監視開始チェックポイントが到達したとき、開始チェックポイント到達フラグを TRUE にするものとする【NWDGM054】。

デッドライン監視開始チェックポイントが到達し、そのチェックポイントがアクティブなモードで参照されており、対応するデッドライン監視開始チェックポイント到達フラグが TRUE のとき、WdgM_CheckpointReached 関数は現在のタイムスタンプと対応するデッドライン監視開始チェックポイントのタイムスタンプの差を計測する。開始チェックポイント到達フラグを FALSE にする【NWDGM055】。

計測された時間の差が WdgMDeadlineMin ～ WdgMDeadlineMax の範囲内でなかった場合、WdgM_CheckpointReached はその監視エンティティを異常とみなす。そうでなければ正常とみなす【WDGM294】。

デッドライン監視開始チェックポイントでも、デッドライン監視終了チェックポイントでもないチェックポイントからの通知を受けたら、WdgM_CheckpointReached はそのチェックポイントを無視し、デッドライン監視の監視結果を更新しない【WDGM299】。

各デッドライン監視開始チェックポイントにおいて、そのチェックポイントが到達したときのタイムスタンプと各監視エンティティのデッドライン監視結果は Watchdog Manager のデバッグで使用可能とする【WDGM241】。

WdgMDeadlineTimeoutDetect が TRUE のとき、WdgM_MainFunction はデッドライン監視開始チェックポイント到達フラグが TRUE であればタイムスタンプを取得し、WdgMDeadlineMax より多くの時間が経過していればデッドライン監視の監視結果を異常とみなす【NWDGM076】。

WdgMDeadlineStartRef および WdgMDeadlineStopRef 参照先のチェックポイントが所属する監視エンティティの WdgMOSCounter 参照先の OsCounterMaxAllowedValue が、 $(WdgMDeadlineMax + WdgMSupervisionCycle * 2) / OsSecondsPerTick$ 以下のコンフィギュレーションをエラーとすること【NWDGM130】。

2.5.3 ロジカル監視

ロジカル監視は監視エンティティのコードが正常にシーケンス通り実行されているかを監視する。

2.5.3.1 ロジカル監視のコンフィギュレーション

全てのロジカル監視は、チェックポイント同士を接続したグラフを有する。

グラフには以下の 2 種類がある。

- ・内部グラフ

グラフ内の全てのチェックポイントが同じ監視エンティティに所属している。

1 つの監視エンティティにつき、0 もしくは 1 つ有することができる。

- ・外部グラフ

グラフ内の少なくとも 1 つのチェックポイントが異なる監視エンティティに所属している。

そのチェックポイントは外部に遷移するよう接続されている。

内部グラフと外部グラフの主な違いは、内部グラフは監視エンティティのプロパティなのに対し、外部グラフはモード依存ということである。

内部グラフのロジカル監視のパラメータは内部遷移(WdgMInternalTransition)であり、それは監視エンティティ内に含まれる。各内部遷移は 2 つのチェックポイントを接続する。これは、全てのモードで同じ内部遷移を共有することを意味している。内部遷移のロジカル監視は監視エンティティを非アクティブにすることで、無効にすることのみ可能である。

外部グラフのパラメータは(WdgMExternalLogicalSupervision)はモード(WdgMMode)に含まれる。各外部遷移は 2 つのチェックポイントを接続する。遷移が接続されているかどうかに関わらず、そのチェックポイントは存在する。

Watchdog Manager はロジカル監視の実施に必要な独立したデータリソースの量を計算する。データリソースの量は監視エンティティ数、WdgMModes の数、WdgMExternalLogicalSupervisions、WdgMInternalTransitions 等から計算する【WDGM293_2】。

2.5.3.2 ロジカル監視のアルゴリズム

Watchdog Manager の初期化直後はまだチェックポイントからの報告はない，すなわち監視エンティティはパッシブである．この情報はアクティブフラグ(1 フラグ/1 監視エンティティ)に保持される．

各内部グラフは 1 つのロジカル監視を表す．N 個の内部グラフがあるとする，監視エンティティは N 個のロジカル監視の監視結果を有することを意味する．

各外部グラフは 1 つのロジカル監視を表すが，それはいくつかの監視エンティティをまたがる可能性がある．監視エンティティをまたぐ M 個の外部グラフがあるとする，M 個のロジカル監視の監視結果が得られることを意味する．

Watchdog Manager は各グラフのアクティブフラグを保持するものとする【WDGM271】．

WdgM_Init 関数は各グラフのアクティブフラグを FALSE にセットする．各グラフは 1 つ，または複数の初期チェックポイントを有することもある．初期チェックポイントとはグラフをスタートすることができるチェックポイントである．チェックポイントの到達を通知するとき，監視エンティティは WdgM_CheckpointReached 関数をコールし，ロジカル監視のアルゴリズムを実行する．遷移が有効であれば，アルゴリズムは直近の到達したチェックポイントを保持しておく必要がある．1 つのチェックポイントは 1 つのグラフにのみ所属することができるため，WdgM_CheckpointReached 関数はそのチェックポイントが所属しているグラフを識別することができる【WDGM296】．

WdgM_CheckpointReached 関数は到達したチェックポイントが所属しているグラフを識別するものとする【WDGM295】．

WdgM_CheckpointReached 関数は各外部グラフと内部グラフの直近に到達したチェックポイントを記憶するものとする．もしグラフのアクティブフラグが TRUE であれば，WdgM_CheckpointReached 関数は新しいチェックポイントが記憶しておいたチェックポイントとの間の遷移であるかどうかと，新しく報告されたチェックポイントが許可されているかどうかをチェックする【WDGM246】．

WdgM_CheckpointReached 関数は以下に従って、内部グラフに所属する報告されたチェックポイントが正常かどうかを確認するものとする【WDGM274】.

- ・アクティブフラグが FALSE のとき

正常：到達したチェックポイントが初期チェックポイント(WdgMInternalCheckpointInitialRef)であるとき

異常：それ以外るとき

- ・アクティブフラグが TRUE のとき

正常：到達したチェックポイントが保存されたチェックポイントの後継であるとき(以下のパラメータ参照)

WdgMInternalTransition,
WdgMInternalTransitionSourceRef,
WdgMInternalTransitionDestRef

異常：それ以外るとき

WdgM_CheckpointReached 関数は以下に従って、外部グラフに所属する報告されたチェックポイントが正常かどうかを確認するものとする【WDGM252】.

- ・アクティブフラグが FALSE のとき

正常：到達したチェックポイントが初期チェックポイント(WdgMExternalCheckpointInitialRef)であるとき

異常：それ以外るとき

- ・アクティブフラグが TRUE のとき

正常：到達したチェックポイントが保存されたチェックポイントの後継であるとき(以下のパラメータ参照)

WdgMExternalTransition,
WdgMExternalTransitionSourceRef,
WdgMExternalTransitionDestRef

異常：それ以外るとき

WdgM_CheckpointReached 関数が正常という結果を得て、かつそのチェックポイントが初期チェックポイントであった場合、WdgM_CheckpointReached 関数はアクティブフラグを TRUE にするものとする【WDGM332】.

WdgM_CheckpointReached 関数が正常という結果を得て、かつそのチェックポイントが最終チェックポイントであった場合、WdgM_CheckpointReached 関数はアクティブフラグを FALSE にするものとする. 最終チェックポイントの後は初期チェックポイントのみ有効である【WDGM331】.

1つのチェックポイントは内部グラフか外部グラフのどちらかに所属することが可能である。これはWDGM274 か WDGM252 の定義でチェックすることを意味する。これは報告されたチェックポイントが任意の内部グラフか外部グラフに所属する場合、WdgM_CheckpointReached 関数がいずれかの実行で、監視エンティティのロジカル監視の結果をセットすることができることを意味している。

報告されてチェックポイントがいずれのグラフにも所属しない場合、ロジカル監視の監視結果は更新されない。これはそのチェックポイントが他の監視機能に利用されるためである。

任意の報告されたチェックポイントがいずれのグラフにも所属していない場合、WdgM_CheckpointReached 関数はそれを無視し、その監視エンティティのロジカル監視の監視結果を更新しない【WDGM297】。

各グラフの直近の到達したチェックポイント、各グラフのアクティブフラグ、各監視エンティティのロジカル監視の監視結果の情報は Watchdog Manager のデバッグに利用可能とする【WDGM242】。

2.6 エラーハンドリング / 復帰

Watchdog Manager は監視の障害から復帰するメカニズムを開始する。これは ECU のグローバルリセットの監視エンティティ内のローカルエラー復帰の範囲である。

2.6.1 RTE モードのメカニズム通知

Watchdog Manager は SW-C と RTE モードメカニズムを介して監視の失敗について SW-C と CDD に通知する。SW-C と CDD は、障害から回復するために、そのアクションをとることができる。

2.6.2 WDGm_GLOBAL_STATUS_STOPPED 時の DEM 報告

Watchdog Manager は状態 WDGm_GLOBAL_STATUS_STOPPED に達したとき、診断イベントマネージャ(Diagnostic Event Manager)にエントリーを登録する。SW-C と CDD はそのエントリーに基づいて復帰を行うことができる。

グローバル監視状態が WDGm_GLOBAL_STATUS_STOPPED、かつ、パラメータ WdgMDemStoppedSupervisionReport が TRUE のとき、Watchdog Manager は DEM にエラー状態を報告するものとする【WDGM129】。

2.6.3 パーティションの再起動 / シャットダウン

Watchdog Manager は非信頼パーティションに配置されている監視エンティティから障害を検出した場合、対応する OSAP を終了することで、再起動/シャットダウンを行うことができる。

監視エンティティのローカル監視状態が WDGm_LOCAL_STATUS_FAILED に変わり、かつ、その監視エンティティが対応する OSAP(WdgMOsApplicationRef)を持つ場合、対応する OSAP の対応するパーティションの再起動/シャットダウン リクエストを送りたい Basic Software Mode Manager の BswM_WdgM_RequestPartitionReset 関数をコールするものとする【WDGM225】。

2.6.4 トリガコンディション設定の停止

グローバル監視状態が WDGm_GLOBAL_STATUS_STOPPED になると Watchdog Manager はトリガコンディションの設定を停止する。その結果 Watchdog タイマがタイムアウトし、ECU がリセットされる【WDGMA006】。

2.6.5 MCU リセット

回復不能な障害が発生したらすぐにマイコンをリセットする必要があるアプリケーション，または，Watchdog タイマから独立したシャットダウンパスを有するアプリケーションに対して，Watchdog Manager は MCU の即時リセットを実行することができる。

パラメータ `WdgMImmediateReset` は `TRUE` に設定されており，かつ，グローバル監視状態が状態 `WDGM_GLOBAL_STATUS_STOPPED` に達した場合，Watchdog Manager は MCU Driver の `Mcu_PerformReset` 関数をコールするものとする．パラメータ `McuPerformResetApi` が `TRUE` でない場合，パラメータ `WdgMImmediateReset` を `TRUE` としないことを推奨する【WDGM133】.

MCU 即時リセットの場合，Watchdog Manager は RTE モードメカニズムを介したアプリケーションへの通知は提供しないものとする【WDGM134】.

2.7 Watchdog のトリガ操作

Watchdog のトリガ操作は Watchdog Manager の重要な役割である。プログラムが正常に動作している間は、Watchdog インスタンスがタイムアウトして ECU がリセットされるのを防ぐ。

通常、Watchdog タイマは独自のタイミング制約を有しており、対応した Watchdog タイマのタイムウィンドウにしたがって周期的にトリガ操作を実施しなければならない。もしトリガ操作を実施しなければ、Watchdog タイマがリセットを行う。

実際のトリガ操作のタイミングは Watchdog Driver 内にカプセル化されている。Watchdog Manager は Watchdog Driver にトリガ操作の継続を Watchdog Interface を介してセットするだけである。

2.7.1 複数 Watchdog 対応

ハードウェアプラットフォームは複数の Watchdog インスタンスを有するよう設計することができるものもある。

Watchdog Manager は複数の Watchdog の並列使用をサポートする【WDGM002】。

2.7.2 トリガコンディション設定

Watchdog Manager は、Watchdog のトリガコンディションを設定するため、Watchdog Interface のサービス WdgIf_SetTriggerCondition を使用する。

このサービスは、Watchdog Driver のインデックス、及びタイムアウト/カウンタをパラメータとして要求する。(WdgMTrigger 参照)

Watchdog Manager は、グローバル監視状態が遷移する際のトリガコンディションの設定は以下のルールに従って行う【NWDGM001】。

1. 状態 WDGM_GLOBAL_STATUS_OK, WDGM_GLOBAL_STATUS_FAILED 及び WDGM_GLOBAL_STATUS_EXPIRED に遷移した際は、関数 WdgM_MainFunction はトリガコンディションを正しく設定する。
2. 状態 WDGM_GLOBAL_STATUS_STOPPED に遷移した際は、関数 WdgM_MainFunction はトリガコンディションを、Watchdog タイマによりリセットさせるため 0 に設定する。
3. 状態 WDGM_GLOBAL_STATUS_DEACTIVATED に遷移した際は、トリガコンディションを設定しない

グローバル監視状態が WDGM_GLOBAL_STATUS_DEACTIVATED に再計算される場合、Watchdog Manager は正しく初期化されていないことを意味し、関数 WdgIf_SetTriggerCondition をコールしないものとする【WDGM292】。

グローバル監視状態が WDMG_GLOBAL_STATUS_OK に再計算される場合、Watchdog Manager は、WDGIF_OFF_MODE に設定されていない全ての Watchdog に対し、ID に WdgMWatchdogDeviceRef, トリガコンディションに WdgMTriggerCondition を設定し、WdgIf_SetTriggerCondition をコールするものとする【WDGM119】.

グローバル監視状態が WDMG_GLOBAL_STATUS_FAILED に再計算される場合、Watchdog Manager は、WDGIF_OFF_MODE に設定されていない全ての Watchdog に対し、ID に WdgMWatchdogDeviceRef, トリガコンディションに WdgMTriggerCondition を設定し、WdgIf_SetTriggerCondition をコールするものとする【WDGM120】.

グローバル監視状態が WDMG_GLOBAL_STATUS_EXPIRED に再計算される場合、Watchdog Manager は、WDGIF_OFF_MODE に設定されていない全ての Watchdog に対し、ID に WdgMWatchdogDeviceRef, トリガコンディションに WdgMTriggerCondition を設定し、WdgIf_SetTriggerCondition をコールするものとする【WDGM121】.

グローバル監視状態が WDMG_GLOBAL_STATUS_STOPPED に再計算される場合、Watchdog Manager は、WDGIF_OFF_MODE に設定されていない全ての Watchdog に対し、ID に WdgMWatchdogDeviceRef, トリガコンディションに 0 を設定し、WdgIf_SetTriggerCondition をコールするものとする. トリガコンディションに 0 を設定すると、Watchdog Driver は直ちに Watchdog タイマへのトリガ操作を止める【WDGM122】.

2.7.3 構成可能パラメータ

Watchdog のトリガ操作のパラメータはモードごとに構成可能である。

2.8 エラー仕様

2.8.1 エラー検出

拡張エラーの検出は、プレコンパイル時(ON / OFF)設定可能とする。パラメータ WdgMDevErrorDetect ([WDGM301_Conf])は、すべての拡張エラーの検出を有効または無効にするものとする【WDGM047】。

WdgMDevErrorDetect が有効になっている場合は、API パラメータのチェックが有効となる。検出されたエラーの詳細な説明は、3.5.2 を参照【WDGM260】。

2.8.2 エラー通知

プリプロセッサで WDGM_DEV_ERROR_DETECT が設定されている場合、検出された拡張エラーを DET の Det_ReportError サービスに通知するものとする【WDGM048】。

2.9 モード切替

2.9.1 監視状態への影響

WdgM_SetMode 関数は異なるモードに変更させるために利用される。モードはコンフィギュレーションにより静的に設定される。

モード切替は監視エンティティの監視パラメータを変更する。

グローバル監視状態が WDGM_GLOBAL_STATUS_OK と WDGM_GLOBAL_STATUS_FAILED でない場合、WdgM_SetMode 関数は何もしないものとする【WDGM316】。

モード切替によって監視エンティティがアクティブからアクティブとなる場合、監視エンティティの現在の状態を保持するものとする【WDGM182】。

モード切替によって監視エンティティがアクティブから非アクティブとなる場合、WdgM_SetMode 関数はその監視エンティティの状態を WDGM_LOCAL_STATUS_DEACTIVATED にする。その際、アライブ監視、デッドライン監視、ロジカル監視それぞれの監視結果を正常とし、異常をカウントするカウンタをクリアする【WDGM315】。

モード切替の際、新しく始めるモードに依存した監視機能(アライブ監視、デッドライン監視、外部ロジカル監視)が管理する変数を全て初期化する【NWDGM085】。

これは、アライブ監視、デッドライン監視、外部ロジカル監視はモードで独立して管理される機能であり、監視エンティティがアクティブからアクティブとなる場合でもモード切替前後の監視機能に関係はないためである。

WdgM_SetMode がコールされた場合、モード切替前後でモードに変化がなかったとしても、モードに依存した監視機能(アライブ監視、デッドライン監視、外部ロジカル監視)が管理する変数を全て初期化する【NWDGM086】。

WdgM_SetMode がコールされた場合、そのタイミングから新しい監視を開始するためである。

2.9.2 Watchdog への影響

WdgM_SetMode 関数がコールされると、Watchdog Manager は、WdgIf_SetMode 関数をコールすることによって各 Watchdog に WdgMWatchdogMode パラメータを設定する【WDGM186】。

WdgM_SetMode は WdgIf_SetMode が正常に完了すると、WdgIf_SetTriggerCondition を用いて、全ての Watchdog に対して新しいモードに対応したトリガコンディション設定を実施する【NWDGM089】。

メモ：

WdgIf_SetMode 関数へのコールが失敗した場合、Watchdog Manager はグローバル監視状態に WDGM_GLOBAL_STATUS_STOPPED に設定する。

2.9.3 スリープ中の Watchdog 操作

ECU State Manager が SLEEP 状態に入るとき、スリープモードを起動させて、WdgM_DeInit 関数をコールする。

WdgM_DeInit 関数は、インテグレータによって定義されるスリープモードへの Watchdog Manager モードの切替を介してトリガコンディションを更新し、Watchdog Manager を非初期化する。

Watchdog リセットなしでシャットダウンの残りを実行可能にするタイムアウトに実行する全ての Watchdog のトリガコンディションを更新するため、モード切替は必要とされている。

ECU が SLEEP 状態にある間、Watchdog Manager は停止する。Watchdog タイマが SLEEP 状態の間、非アクティブ状態にできない、あるいはしないものとする場合、必然的に Watchdog リセットに繋がる。

BSW コンポーネントは、Watchdog Manager が非アクティブである間、Watchdog タイマのトリガ操作をする場合がある。

インテグレータは、必要に応じてモードを設定する

2.10 ポートとポートインタフェースの仕様

この章では、Watchdog Manager によって提供された AUTOSAR インタフェースを定義する。

Watchdog Manager サービスの SW-C の記述は、SW-C と CDD が利用可能な Watchdog Manager ポートを定義する。

サービスを利用する各 AUTOSAR SW-C または CDD は、それら自身の記述内にサービスポートを含める必要がある。

RTE が適切な ID および要求されるシンボルを生成できるように、これらのポートは、同じインタフェースに分類され、Watchdog Manager のポートに接続される必要がある。

Watchdog Manager のローカル監視状態とグローバル監視状態はモードポートを介して SW-C および CDD へ報告されている。

SW-C と CDD は、Watchdog Manager のモードポートと同じインタフェースを持つ独自のモードポートを定義できる。

その後、SW-C または CDD はステータスを通知でき、そしてモードポートを経由して状態の変化が通知される。

加えて、SW-C は状態変化で RTE によって開始または停止されるランナブルを定義できる。

BSW モジュールは API 関数を直接コールし、RTE によるマッピングを考慮しなければならない。

インタフェース定義の全ては、以下で解釈される。

ARPackage AUTOSAR/Services/WdgM

2.10.1 アライブ監視のためのポートとポートインタフェース

2.10.1.1 一般的なアプローチ

SW-C と CDD とサービスの間の全てのインタフェースは、Watchdog Manager によって提供されるポートの数を減らすために、クライアント/サーバ通信としてモデル化される。

チェックポイントをレポートするためには、セNDER/レシーバ パラダイムがより適切に見えるかもしれないが、この主のモデリングはポートの数を倍増させることが考えられる。

そのためにも、クライアント/サーバ パラダイムは選択されている。

固有な監視エンティティ ID は ECU 内の監視エンティティを識別するための使用される。

アプリケーションコードを、ECU 依存な監視エンティティ ID のコンフィギュレーションから独立し続けるために、SW-C および CDD によって使われる ID は、SW-C とサービスの間で渡されるデータ要素のような明示的なモデル化はされていない。

これらの ID は、Watchdog Manager の提供側ポートの「ポート定義された引数の値」としてモデル化される。

その結果、監視エンティティ ID はクライアント/サーバ インタフェース操作の引数として表示されることはない。

このアプローチの更なる結果として、各監視エンティティに個別のポートが存在することになる。

2.10.1.2 データタイプ

アプリケーションとサービスの間で渡されたパラメータは以下のとおり

1. 監視エンティティを識別する ID
2. チェックポイントを識別する ID

この監視エンティティ ID のための型は WdgM_SupervisedEntityType 型をベースとしなければならない。この型は uint16 として定義される。

そのため、以下の型定義が必要となる。

```
ImplementationDataType WdgM_SupervisedEntityType {  
    CATEGORY = TYPE_REFERENCE;  
    implementationDataType = uint16/uint8;  
};
```

すべての Watchdog Manager API は、uint16 の代わりに WdgM_SupervisedEntityType のタイプを使用している。

効率化のために、整数タイプが ECU 間で変化することがある。

したがって、ECU の特定のタイプ定義は、「AUTOSAR タイプ」を提供する代わりに Watchdog Manager で提供されなければならない。

このチェックポイント ID のタイプは、WdgM_CheckpointIdType のタイプに基づく。

このタイプは、uint8 か uint16 で定義される

しかたがって、タイプは以下に従う。

```
ImplementationDataType WdgM_CheckpointIdType {  
    CATEGORY = TYPE_REFERENCE;  
    implementationDataType = uint16/uint8;  
};
```

チェックポイント ID の範囲は、すべての構成された監視エンティティとプラットフォーム対応の上に依存する。

2.10.1.3 アライブ監視のためのポートインタフェース

XML 記述に必要なポートと名前を最小限にするために、全ての操作は 1 つのインタフェース内に配置される。

したがって、本仕様書内で定義された API と一致した以下の操作を持つことになる。

```
ClientServerInterface WdgM_AliveSupervision {  
    isService = true;  
    PossibleErrors {  
        E_NOT_OK  
    };  
    UpdateAliveCounter (ERR{E_NOT_OK});  
    CheckpointReached(IN WdgM_CheckpointIdType,  
        ERR{E_NOT_OK});  
};
```

API と比較すると、その名前はインタフェース記述の一部として XML でグローバルではない状態で表示されるため、名前に「WdgM_」というプレフィックスは不要となる【WDGM333】。

2.10.1.4 サービスポート

Watchdog Manager 側では、上記の WdgM_AliveSupervision のインタフェースの全てのサービスを提供する監視エンティティごとに 1 つのポートがある。

各監視エンティティは、そのアプリケーションに関連付けられた監視エンティティごとに、それらのサービスが要求する 1 つのポートを持つ。

Watchdog Manager は構成された各監視エンティティのアライブ監視のために単一のサービスポートを提供するものとする。状態通知のためにアライブ監視ポートとそれに対応するモードポートが一致できるようには、命名規則が必要になる【WDGM146】。

Watchdog Manager のアライブ監視ポートは、”alive000”, “alive001”... “alive<#SE-1>”と命名されなければならない。その番号は 0 から始まり、構成された監視エンティティの数に達するまでの連続したものとする【WDGM147】。

2.10.1.5 エラーコード

アライブ監視のサービスは、サービス固有のエラーコードを返さない。

2.10.2 状態通知のためのポートとポートインタフェース

2.10.2.1 一般的なアプローチ

SW-C 及び CDD の動作を、状態に伴って制御させるために、RTE はモードポートのメカニズムを提供する。

モードマネージャは、モードポート内で定義された異なるモード間で、切り替えことができる。

モードポートに接続された SW-C/CDD は、2 つの方法で、モード情報を利用することができる。

- SW-C/CDD は、モードポートを介して、現在モードを問合せることができる。
- SW-C/CDD は、モード変化により、RTE から開始または停止されるランナブルを宣言することができる。

RTE 仕様書によると、モードポートは、「ModeSwitchInterface」を持つ。

モードマネージャ、ここでは Watchdog Manager は送信側であり、SW-C は受信側である。

Watchdog Manager は、2 種類の情報を提供するため、モードポートを利用する。

- 第一に、それは、各監視エンティティのローカル監視状態を提供する。
- 第二に、Watchdog Manager は、全監視エンティティの監視状態の集約を反映した、グローバル監視状態を提供する。よって、それは特別なモードポートを 1 つ持つ。

Watchdog Manager のモードポートは、以下のモードを宣言するものとする【WDGM195】。

- STATUS_OK
- STATUS_FAILED
- STATUS_EXPIRED
- STATUS_STOPPED
- STATUS_DEACTIVATED

この定義は、「WdgM_LocalStatusType」型に対応する。

Watchdog Manager は、状態変化発生時、RTE のモードポートを通じて、SW-C/CDD に通知する。通知にあたり、直接的または間接的のどちらの RTE の API を使用するかは、実装依存である【WDGM196】。

直接的 API を介した実装においては、それぞれの監視エンティティ用に生成された API をコールしなければならない。

```
Rte_StatusType Rte_Switch_mode<SEID>_currentMode(  
    Rte_ModeType_WdgMMode mode  
)
```

グローバル状態用

```
Rte_StatusType Rte_Switch_globalMode_currentMode(  
    Rte_ModeType_WdgMMode mode  
)
```

「mode」となっているところは、通知される新しいモードである。値の範囲は、前の要求(SWS_WdgM_00195)にて定義されている。戻り値は、無視することができる。

「2.11.2.3 ポートインタフェース」にあるような、間接的なポート API を使用方法は、それぞれの監視エンティティに状態通知する場合に、上記 API の代わりに使用することで、少ないコードで実現できる。

監視エンティティのローカル監視状態が変化した場合、Watchdog Manager は、それを認識でき次第すぐ、その監視エンティティのモードポートを介して、その変化を通知する【WDGM197】。

グローバル監視状態が変化した場合、Watchdog Manager は、グローバルのモードポートを介して、その変化を通知する【WDGM198】。

全てのローカル監視状態に基づいてグローバル監視状態を割り出した後、Watchdog Manager は、グローバル監視状態を導出する際の様々な変化を、1 回だけ通知する。結果的に動作は、最初に、ローカル監視状態の全ての変化が通知される。つぎに、それぞれ(ローカル監視状態)の変化により、それ(グローバル監視状態)が変化した場合に、グローバル監視状態が 1 回だけ通知される。例は以下に示す【WDGM199】。

例)

① 1 監視サイクル内で、

SE1 が「OK」から「FAILED」に遷移したとき、

SE1 の個別のモードポートで「FAILED」が通知される。

② 同一監視サイクル内で、

SE2 が「OK」から「EXPIRED」に直接遷移したとき、

SE2 の個別のモードポートで「EXPIRED」が通知される。

③ その監視サイクル内で、

グローバル監視状態は結果的に「OK」から「EXPIRED」に変化することになり、

「EXPIRED」のみがグローバルのモードポートで通知される。

この例中の、「FAILED」はグローバルのモードポートで通知されない。

なぜならば、それは一組の監視エンティティの状態確認のさなかにおける単なる中間の状態であるからである。

2.10.2.2 データタイプ

モード宣言グループ「WdgMMode」は、SW-C/CDD 及び RTE に通知される Watchdog Manager のモードを表す。このモードの定義は「WdgM_LocalStatusType」型に対応する。

WdgMMode は以下のとおりである【WDGM334】。

```
ModeDeclarationGroup WdgMMode {  
    {  
        SUPERVISION_OK,  
        SUPERVISION_FAILED,  
        SUPERVISION_EXPIRED,  
        SUPERVISION_STOPPED,  
        SUPERVISION_DEACTIVATED  
    }  
    initialMode = SUPERVISION_OK  
};
```

2.10.2.3 ポートインタフェース

関係する SW-C/CDD 及び RTE に対して、監視状態の変化を通知するための 2 つの異なるインタフェースが存在する。

インタフェース「WdgM_IndividualMode」は、ひとつの監視エンティティのローカル監視状態を通知するために使用される。

WdgM_IndividualMode は以下のとおり【WDGM335】。

```
ModeSwitchInterface WdgM_IndividualMode {  
    isService = true;  
    WdgMMode currentMode;  
};
```

インタフェース「WdgM_GlobalMode」は、全ての個別の監視エンティティの集約であるグローバル監視状態を通知するために使用される。

WdgM_GlobalMode は以下のとおり【WDGM336】.

```
ModeSwitchInterface WdgM_GlobalMode {  
    isService = true;  
    WdgMMode currentMode;  
};
```

2つの異なったインタフェースを定義する理由としては、これらのインタフェースの使われ方にある。

「WdgM_GlobalMode」インタフェース用に、Watchdog Manager は、唯一つの単一ポートを提供する。

それとは対照的に、「WdgM_IndividualMode」インタフェース用に、Watchdog Manager は、監視エンティティの数分だけポートを提供する。

それらのポートに効率的にアクセスするために、間接的なポート API が使用される。

この API は、同じインタフェースを持つ全てのポートのリストを提供する。

例えば、

```
/* WdgM 内でコールされる */  
/* 監視エンティティの状態/モードを、RTE を通じて、SW-C/CDD に通知 */  
void WdgM_NotifyOKToSE(WdgM_SupervisedEntityType se)  
{  
    Rte_PortHandle_WdgM_IndividualMode_P ph =  
        Rte_Ports_WdgM_IndividualMode_P0;  
    ph[se].Switch_currentMode(RTE_MODE_WdgM_Mode_SUPERVISION_OK);  
}
```

グローバル監視状態のモードポートが、このリストに出現してしまうことを避けるために、このポートは異なったインタフェース、すなわち「WdgM_IndividualMode」の代わりに「WdgM_GlobalMode」を使用する。

2.10.2.4 モードポート

Watchdog Manager は、設定されている各監視エンティティのローカル監視状態を通知するための単一なモードポートを提供する。アライブ監視ポートと、対応する状態通知用のモードポートの一致をとるために、命名規則が必要である【WDGM148】。

Watchdog Manager の、各監視エンティティの監視状態を通知するための単一モードポートは、"mode000", "mode001" ... "mode<#SE-1>"と命名される。番号は 0 から始まり、設定した監視エンティティの数に達するまで連続する。さらに、Watchdog Manager は、グローバル監視状態を通知できるようにする【WDGM149】。

Watchdog Manager は、グローバル監視状態を通知するためのモードポートを、1 つ提供する【WDGM150】。

2.10.2.5 エラーコード

モードポートは、エラーを通知することができない。

内部動作

まず、サービスのランナブルエンティティは、「内部動作(Internal Behavior)」の記述の中で指定されなければならない。このサービスの記述に関連するランナブルエンティティは、アプリケーション SW-C からアクセスされる BSW モジュールのサービスを果たす API である。

その導出が、以下の記述である。

```
// Runnable entities of the Watchdog Manager
RunnableEntity SetMode
    symbol "WdgM_SetMode"
    canbeInvokedConcurrently = FALSE
RunnableEntity GetMode
    symbol "WdgM_GetMode"
    canbeInvokedConcurrently = FALSE
RunnableEntity CheckpointReached
    symbol "WdgM_CheckpointReached"
    canbeInvokedConcurrently = TRUE
RunnableEntity UpdateAliveCounter
    symbol "WdgM_UpdateAliveCounter"
    canbeInvokedConcurrently = TRUE
```

その後、「内部動作(Internal Behavior)」は、アライブ監視ポートに対して、ポート定義引数値を定義する。

```
PortArgument{port= alive000, value.type= WdgM_SupervisedEntityType, value.value=0};
PortArgument{port= alive001, value.type= WdgM_SupervisedEntityType, value.value=1};
...
PortArgument{port= alive<#SE-1>, value.type= WdgM_SupervisedEntityType,
value.value=<#SE-1>;
```

そして最後に、「内部動作(Internal Behavior)」は、状態通知するモードポートに間接アクセスするための特別な API の生成を行うよう、RTE に指示する。

```
IndirectAPI{port= mode000};
IndirectAPI{port= mode001};
...
IndirectAPI{port= mode<#SE-1>;
```

2.10.3 Watchdog Manager のサービスの定義

このセクションでは、Watchdog Manager サービスの一通りの定義の例を示す。

これらの定義は、ECU コンフィギュレーションの間だけで、完結してしまうことに注意。

(なぜならば、これらの定義は、Watchdog Manager サービスにより提供されるポート数を決定する Watchdog Manager パラメータの特定の設定に依存するため。)

また、SW-C/CDD の実装は、これらの定義には依存しないことに注意。

RTE には（上下）両面にポートが存在するが、ここでの Watchdog Manager サービスの記述は、RTE の下側のポートを定義する。

サービスを利用する各 SW-C/CDD は、Watchdog Manager モジュールのポートに接続される「サービスポート」を、自身の SW-C/CDD の記述の中に含む必要があり、それによって RTE が生成される。

これは Watchdog Manager のサービスの記述である。

これは Watchdog Manager の外観であり、SW-C/CDD/ECU インテグレータに対して可視化されなければならない。

以下に Watchdog Manager サービスを示す【WDGM338】

```
/** This is the definition of the Watchdog Manager as a
 * service. This is the outside view of the Watchdog Manager,
 * which must be visible to the SW-Cs/CDDs / ECU integrator
 **/

Service WdgM {
// For each supervised entity the Watchdog Manager
// provides a port to update the alive counter
ProvidePort WdgM_AliveSupervision alive000;
...
ProvidePort WdgM_AliveSupervision alive<#SE-1>;

// For each supervised entity the Watchdog Manager
// provides a mode port to signal the Local
// Supervision Status to interested SW-Cs/CDDs and the RTE
ProvidePort WdgM_IndividualMode mode000;
...
ProvidePort WdgM_IndividualMode mode<#SE-1>;

// The Watchdog Manager also provides a single mode port
```

```
// to signal the Global Supervision Status to
// interested SW-Cs/CDDs and the RTE
ProvidePort WdgM_GlobalMode globalMode;

InternalBehavior
{
    // Runnable entities of the Watchdog Manager
    RunnableEntity UpdateAliveCounter
        symbol "WdgM_UpdateAliveCounter"
    canbeInvokedConcurrently = TRUE

    RunnableEntity CheckpointReached
        symbol "WdgM_CheckpointReached"
    canbeInvokedConcurrently = TRUE

    PortArgument{port= alive000, value.type=
WdgM_SupervisedEntityIdType, value.value=0};

    PortArgument{port= alive001, value.type=
WdgM_SupervisedEntityIdType, value.value=1};

    ...

    PortArgument{port= alive<#SE-1>, value.type=
WdgM_SupervisedEntityIdType, value.value=<#SE-1>};

    IndirectAPI{port= mode000};

    IndirectAPI{port= mode001};

    ...
    IndirectAPI{port= mode<#SE-1>};
};
};
```

2.11 関連モジュール

Watchdog Manager と他のモジュールとの関係を以下の図に示す。

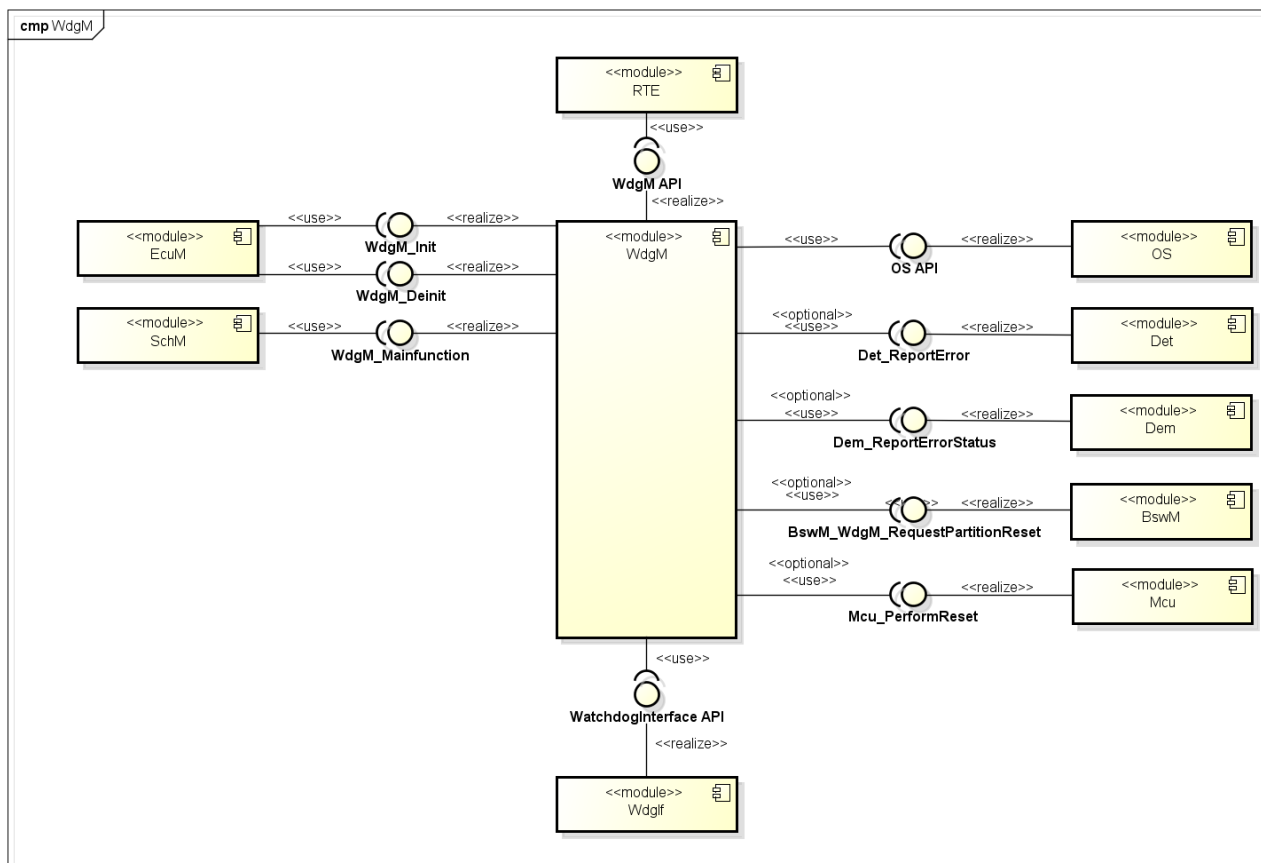


図 2.11-1：他モジュールとの関係

2.11.1 Runtime Environment (RTE)

RTEはSW-CやCDD内の監視エンティティから Watchdog Manager にチェックポイント情報を伝えることに対して責任がある。Watchdog Manager は監視状態の変更について RTE のサービスを使用して SW-C に通知する。BSWM は RTE を使わずに Watchdog Manager をコールすることができる。

2.11.2 Watchdog Interface (WdgIf)

Watchdog Manager は Watchdog Driver のモード切替および Watchdog タイマへのトリガ操作のコンディショニングを Watchdog Driver に報告することに対して責任がある。Watchdog Driver は複数の Watchdog インスタンスをアドレス指定することを許可する Watchdog Interface を通してアドレス指定される。

2.11.3 ECU State Manager (EcuM)

ECU State Manager は Watchdog Manager の初期化、非初期化およびスリープモード時の Watchdog タイマのトリガ操作に対して責任がある。

2.11.4 BSW Scheduler (SchM)

BSW Scheduler は Watchdog Manager の周期関数のコールに対して責任がある。Watchdog Manager は重大なセクションの実装のために BSW Scheduler のサービスを使用する。

2.11.5 OS

OS は以下機能の Watchdog Manager への提供に対して責任がある。

- ・ OS アプリケーションの状態をアクセス許可に設定
- ・ 信頼機能のコール
- ・ ISR メモリのアクセス許可状態取得
- ・ オブジェクト (Task や ISR 等) のアクセス許可状態取得
- ・ オブジェクト (Task や ISR 等) の所属する OS アプリケーション情報の取得
- ・ タスクメモリのアクセス許可状態取得
- ・ 実行中の OS アプリケーション ID の取得
- ・ OS アプリケーションの状態取得
- ・ 前回コールからの差分カウンタ値取得
- ・ 実行中の ISR ID の取得
- ・ スケジュールテーブルの状態取得
- ・ ソフトウェアカウンタのインクリメント
- ・ スケジュールテーブルの切替
- ・ スケジュールテーブルの同期停止
- ・ 下層のカウンタ値が絶対値”Start”でスケジュールテーブルの処理を開始
- ・ 下層のカウンタ値が値”Now”に関連した”Offset”でスケジュールテーブルの処理を開始
- ・ 他スケジュールと同期してスケジュールを開始
- ・ 実行中のスケジュールテーブルの即時停止
- ・ スケジュールテーブルの同期開始と同期カウンタの提供

2.11.6 BSW Mode Manager (BswM)

BSW Mode Manager は非信頼パーティションの再起動に対して責任がある。監視エンティティは OS アプリケーションに関連付けることができる。監視エンティティの監視が失敗したとき、Watchdog Manager は関連したパーティションの再起動を要求する。

2.11.7 Micro Controller Unit Driver (Mcu)

Watchdog Manager は監視異常の場合に、ECU の即時リセットをしてもよい。リセットサービスは MCU Driver によって提供される。

2.11.8 Development Error Tracer (Det)

開発エラー検出が有効の場合、Watchdog Manager は検出された開発エラーについて DET に報告する。

2.11.9 Diagnostic Event Manager (Dem)

Watchdog Manager は検出された標準/拡張エラーを DEM に通知してもよい。

2.12 ファイル構成

本仕様ではソースファイル構成を完全には定義しないものとする。

2.12.1 ソースファイル構成

本節では、特筆すべきファイルのみ記載する。

Watchdog Manager のソースファイルには、以下のファイルが含まれる。

以下のソースファイルには

表 2-1 ソースファイル

WdgM_PBcfg.c	実装ソースファイル。 ポストビルドコンフィギュレーションを使用する場合存在する。ポストビルドコンフィギュレーションをこのファイルに含む【WDGM127】。
WdgM_Cfg.c	プレコンパイルコンフィギュレーションのうち、型を持つインスタンスとなる定数、変数を定義する【NWDGM112】。
WdgM.c	システムサービスの実装ファイル【NWDGM113】

2.12.2 ヘッダーファイル構成

本節では、特筆すべきファイルのみ記載する。

Watchdog Manager のヘッダーファイルには、以下のファイルが含まれる。

表 2-2 ヘッダーファイル

Dem.h	エラーや必須のイベント ID を報告する API をインクルードするものとする。この仕様は XML によって DEM 構成ツールに提供されるイベント ID の名前を定義する。DEM 構成ツールは ECU 依存な値をイベント ID シンボルに割当て、em_IntErrId.h でシンボルを提供する【WDGM126】。
Rte_WdgM_Type.h	WdgM.h ファイルが BSW モジュールとソフトウェアコンポーネントによって使われた共通であるタイプをインクルードするためにインクルードする。WdgM.h ファイルは Rte_WdgM_Type.h を介してインクルードされ、またはまだ定義されていない型のみを含むものとする【WDGMa005】。
WdgM.h	外部モジュールに提供する API、定数、型を含むものとする【NWDGM110】。
WdgM_Cfg.h	プレコンパイルコンフィギュレーションのうち、マクロ定義の定数を定義する【NWDGM111】。

ファイルのインクルード関係は下図に従うものとする【WDGM014】.

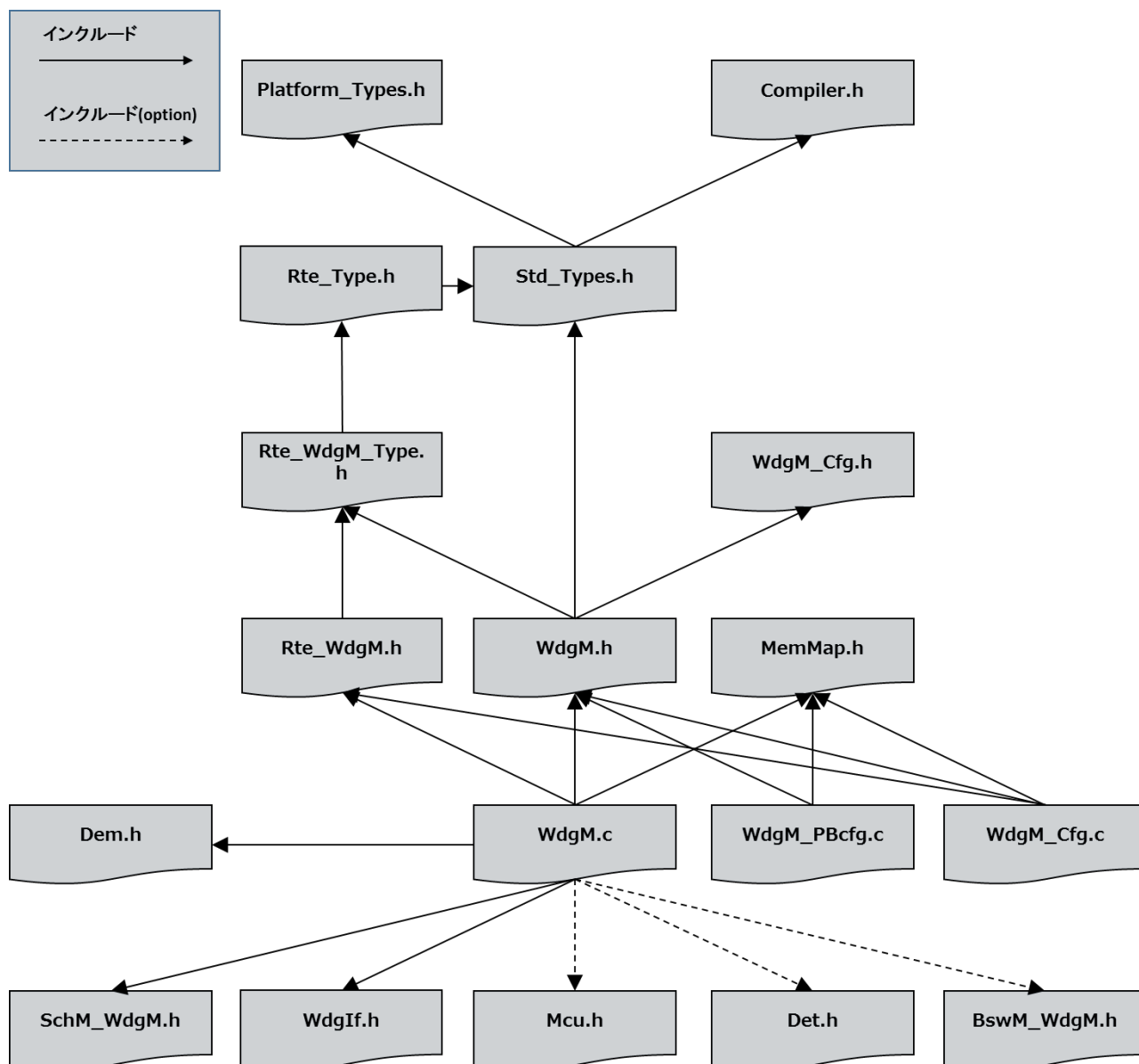


図 2.12-1 : インクルード構成

2.13 コンフィギュレーション

2.13.1 コンフィギュレーションクラス

VARIANT-PRE-COMPILE はプレコンパイルタイムでのみコンフィギュレーション可能なパラメータであることを示す【WDGM265】.

VARIANT-POST-BUILD はポストビルドタイムまたはプレコンパイルタイムにコンフィギュレーション可能なパラメータであることを示す【WDGM266】.

Watchdog Manager ではポストビルドタイムのパラメータはセレクトブルのみ対応するものとする【NWDGM052】.

2.13.2 コンフィギュレーションルール

システムジェネレーションまたはシステムコンパイル時に最小限に構成されるべき Watchdog Manager のパラメータはモジュールのコンフィギュレーションヘッダーファイルである WdgM_Cfg.h に置くべきである【WDGM025】.

ポストビルドタイムで構成されるべき Watchdog Manager のパラメータは WdgM_ConfigType 型の外部データ構造に置かれるべきである【WDGM029】.

プレコンパイルオプションはコードジェネレータで直接生成されないコード実装で利用されるべきである. それ故に, プレコンパイルオプションは静的なコンフィギュレーションの設定に従い Watchdog Manager の再利用ソースコードの最適化を補助する. それらはモジュールのコンフィギュレーションのヘッダーファイルである WdgM_Cfg.h に置かれなければならない【WDGM104】.

2.13.3 公開パラメータ

“SRS General on Basic Software Modules”の BSW00402 で要求されている標準化された共通の公開パラメータは Watchdog Manager のヘッダーファイル内で公開するものとし, また, ”BSW Module Description”内で提供される必要がある. 関係するモジュールの略称は”List of Basic Software Modules”に記載されている【WDGM267】.

2.13.4 モード非依存監視設定

2.13.4.1 監視エンティティ

プラットフォームを跨った SW-C の移植性をサポートするために、Watchdog Manager は ECU 毎の監視エンティティの量に順応させる必要がある。

各監視エンティティ固有の監視エンティティ ID はコンフィギュレーションパラメータ WdgMSupervisedEntityID によって供給される。Watchdog Manager の範囲内では固有の ID とすること【WDGM304】。

各 BSWM は自身のモジュール ID を監視エンティティ ID として使用すること【WDGM306】。

BSWM が配置されているかどうかに関わらず、SW-C は監視エンティティ ID として如何なる BSWM の ID も使用しないこと【WDGM305】。

Watchdog Manager コンフィギュレーションジェネレータは監視エンティティ ID が固有でないコンフィギュレーションを拒否すること。監視エンティティやチェックポイントはモードに関わり無く存在する。他方で監視機能は部分的にモードに関係なく存在していたり、部分的にモードに依存していたりする【NWDGM060】。

与えられたモード内で監視状態がアクティブな監視エンティティを持つために、以下を満たすこと【WDGM282】。

- 1.監視エンティティはモードから参照されること。(WdgMMode -> WdgMLocalStatusParams -> WdgMLocalStatusSupervisedEntityRef -> WdgMSupervisedEntity)
- 2.監視機能のモード依存のセッティングの内少なくとも一つは、与えられたモードのために設定されること。

与えられたモード内で監視状態が非アクティブな監視エンティティを持つために、以下を満たすこと【WDGM283】。

- 1.監視エンティティはモードから参照されないこと。(WdgMMode -> WdgMLocalStatusParams -> WdgMLocalStatusSupervisedEntityRef -> WdgMSupervisedEntity)
- 2.監視機能のモード依存のセッティングが一つも、与えられたモードのために設定されないこと。
内部グラフのロジカル監視が監視エンティティのプロパティであるため、内部グラフのロジカル監視のコンフィギュレーションは、監視エンティティのアクティブ/非アクティブ状態に影響を与えない。

Watchdog Manager コンフィギュレーションジェネレータは以下を全て満たすコンフィギュレーションをエラーとすること。【NWDGM114】

- 1.監視エンティティがモードから参照される。(WdgMMode -> WdgMLocalStatusParams -> WdgMLocalStatusSupervisedEntityRef -> WdgMSupervisedEntity)

2.監視機能のモード依存のセッティングが1つも，与えられたモードのために設定されない．

Watchdog Manager コンフィギュレーションジェネレータは以下を全て満たすコンフィギュレーションをエラーとすること．【NWDGM115】

1.監視エンティティがモードから参照されない．（WdgMMode -> WdgMLocalStatusParams -> WdgMLocalStatusSupervisedEntityRef -> WdgMSupervisedEntity）

2.監視機能のモード依存のセッティングが1つ以上，与えられたモードのために設定される．

2.13.4.2 OS アプリケーション

監視エンティティは信頼/非信頼パーティションに所属できる。各非信頼パーティションは自身の異常が MCU の他パーティションのメモリを破壊しないために隔離されたメモリアクセスを持つ。パーティションは独立して終了や再起動ができる。各パーティションは AUTOSAR OS によって管理される OS アプリケーションと 1 対 1 で一致する。

監視エンティティが構成された OS アプリケーションを持っていた場合、WdgM は監視エンティティのローカル監視状態が `WDGM_LOCAL_STATUS_FAILED` に到達した時、一致するパーティションのリスタートを要求する。

パーティションの再起動を有効にするため、監視エンティティは OS アプリケーション (WdgMOsApplicationRef 参照) を参照する必要がある。OS アプリケーションは非信頼でなければならない。

Watchdog Manager コンフィギュレーションジェネレータは WdgMOsApplicationRef が信頼 OS アプリケーションを参照しているコンフィギュレーションを拒否しなければならない【WDGM311】。

2.13.4.3 内部グラフのロジカル監視

各監視エンティティは Watchdog Manager によって監視される構成された制御フローを持つことができる。この制御フローはチェックポイントと遷移によって抽象化される。チェックポイントの内 1 つが初期チェックポイントとして示される。

Watchdog Manager コンフィギュレーションジェネレータは監視エンティティ内の内部遷移 (WdgMInternalTransition 参照) が接続するチェックポイントの両方が同じ監視エンティティに所属していなかった場合、拒否すること【WDGM212_1】。

モードに依存して内部グラフのロジカル監視の ON/OFF 切替をするために、各モードから監視エンティティが参照 (WdgMLocalStatusParam 参照) される必要がある【WDGM212_2】。

監視エンティティ毎に 0 個または 1 個の内部グラフをもつことができる。さらに、監視エンティティのチェックポイント全てがモニタ必要というわけではない。しかしながら、一つより多くのグラフに所属するチェックポイントはない。これは、各グラフが同時に実行することができることを前提とし、重複の場合には、与えられたチェックポイントが所属するグラフを区別する手段がない【NWDGM132】。

Watchdog Manager は如何なるモードにおいても外部グラフと内部グラフの両方に所属したチェックポイントがあるコンフィギュレーションを拒否すること【WDGM309】。

内部遷移と内部グラフは監視エンティティのプロパティである。これらの内部遷移は監視エンティティの制御フローのみに依存する。それ故に、監視エンティティを含む SW-C や BSWM の開発者は他監視エンティティとは無関係にチェックポイントと内部遷移のコンフィギュレーションを実現することができる。

2.13.5 モード依存パラメータ

2.13.5.1 モード

Watchdog Manager のモードを変更することは、Watchdog 操作のコンディションの変更にも繋がる。従って、Watchdog Manager は各構成されたモードと各 Watchdog のいくつかの静的に構成された Watchdog パラメータを提供する。

各 Watchdog インスタンスのために、Watchdog モードはパラメータ WdgMWatchdogMode によって静的に構成されるものとする【WDGM181】。

Watchdog モードを WDGIF_OFF_MODE にすることで、監視を行わなくすることができる。

Watchdog Manager は、設定されたモード(WdgMMode)と監視エンティティごとに静的に構成された監視パラメータを持つ。

2.13.5.2 外部グラフのロジカル監視

監視エンティティの境界を横切る遷移がある。

Watchdog Manager が複数の監視エンティティの実行シーケンスを監視するとき外部遷移をする。

このように、外部遷移は内部遷移とそのロジカル監視の関連から独立して構成されなければならない。

内部遷移の構成と外部遷移の構成には大きな差がある。

内部遷移は1つの監視エンティティに属しており、それは Watchdog Manager モードに依存しない。これはモードからその監視エンティティが参照されるかどうかによって、アクティブか非アクティブかを構成することができる。しかし、同じ監視エンティティ内で異なる遷移なチェックポイントを持たせることはできない。一方、外部遷移は特定の Watchdog Manager モードに含まれる。モードごとに外部遷移グラフを有することができる。2つの異なるモードが同じグローバル遷移のグローバルグラフを有する場合、それらは重複される必要がある。

Watchdog Manager はいくつかの異なる操作のモード(WdgMConfigSet)をサポートする。

各モードは以下によって定義される。

- ・アクティブな監視エンティティとそれらのパラメータの設定
- ・監視機能
- ・トリガコンディション設定を行う Watchdog の設定

異なるモードは ECU ライフサイクルの異なるフェーズのために必要とされる。例えば、あるモードはスタートアップとシャットダウンの間、別のモードは通常の操作中、さらにまた別のモードはスリープ中にそれぞれアクティブとなる。通常の動作中だけでも複数のモードを必要とされることもある。例えば、複数のアプリケーションが同じ ECU で動作するとき、あるアプリケーションはシャットダウンしていて監視の必要がなく、もう一つのアプリケーションはまだ動作中で監視の必要があるときが考えられる。

各 Watchdog モードは固有な ID(WdgMModeId)を持つものとする 【WDGM178】。

Watchdog Manager は起動時の初期モード(WdgMInitialMode)を持つものとする 【WDGM179】。

外部グラフはオーバーラップすることができない。

Watchdog Manager は1つのチェックポイントが同じモード内では複数の外部グラフに依存する構成を拒否するものとする 【WDGM310】。

2.13.5.3 アライブ監視

各チェックポイントのタイミング制約は Wactchdog Manager の設定可能なパラメータで表現される (WdgMAliveSupervision)。タイミング制約はチェックポイントに定義されるが、Wactchdog Manager で監視エンティティ全体のアライブ監視結果を決定する。

失敗した監視参照サイクルの許容回数は監視エンティティのアプリケーションコンテキストに基づく。したがって、この閾値が該当する監視エンティティが最後に失敗した場合、設定可能なパラメータである必要があるかどうかをチェックする (WdgMFailedSupervisionRefCycleTol)。

アライブ監視がローカル監視状態によって失効した状態に達した時、これはリカバリできない。結果として、Wactchdog のトリガは停止する。しかし、この状態でさらなる反応のために一定時間確保し、Wactchdog のトリガの停止は連続的な監視サイクルの量だけ延期することができる (WdgMExpiredSupervisionCycleTol)。

2.13.5.4 デッドライン監視

Wactchdog Manager のコンフィギュレーションジェネレータは、WdgMDeadlineSupervision パラメータに、WdgMDeadlineStartRef と WdgMDeadlineStopRef パラメータ両方のチェックポイントが所属していない構成を拒否する【WDGM313】。

Wactchdog Manager のコンフィギュレーションジェネレータは、2 つのチェックポイントが複数の WdgMDeadlineSupervision パラメータにある構成を拒否する【WDGM314】。

2.14 自己監視機能

2.14.1 WdgM_MainFunction コール周期/OS カウンタ監視機能

本 Watchdog Manager は WdgM_MainFunction のコール周期の逸脱，または，OS カウンタの不具合を検出する機能を有する．

本 Watchdog Manager が備える WdgM_MainFunction コール周期/OS カウンタ監視機能は以下のようなものである．

- ・ 本監視は WdgM_MainFunction で周期的に実施する【NWDGM094】.
- ・ コンテナ WdgMMFCycleAndOsCounterSupervisionEnabled が TRUE のとき有効となる(デフォルトで無効)【NWDGM095】.
- ・ コンテナ WdgMMFCycleAndOsCounterSupervision が設定されていないモードであれば，本監視は実施しない【NWDGM096】.
- ・ コンテナ WdgMMFCycleAndOsCounterSupervision のパラメータ WdgMOsCounterRef 全てから参照する OS カウンタからタイムスタンプを取得し，WdgM_MainFunction がコールされる間隔を計測する【NWDGM097】.
- ・ 計測された間隔が WdgMMFCycleMinTol ～ WdgMMFCycleMaxTol の範囲内でなかった場合，WdgM_MainFunction のコール周期を異常と判定する【NWDGM098】.
- ・ 本監視の監視結果が異常である場合，WDGM_E_MFCYCLE_OR_OSCOUNTER として DEM を発行し，グローバル監視状態を WDGM_GLOBAL_STATUS_STOPPED に遷移する【NWDGM099】.
- ・ 本監視は WdgM_MainFunction のコール周期か OS カウンタのどちらかが異常であれば異常を検出するが，どちらの異常であるかは区別しない【NWDGM100】.
- ・ 監視モード切替後，最初の WdgM_MainFunction では本監視は実施しない【NWDGM101】.

3 API 仕様

3.1 API 実装方針

3.1.1 実装言語

システムサービスのインタフェースは ISO/ANSI-C に基づいて定義する。

通常はシステムサービスを関数として実装するが、必要に応じて C 言語のプリプロセッサマクロで実装しても良い。

3.1.2 再入可能性

Watchdog Manager API の再入可能性については、各 API の仕様に準ずる。

3.1.3 デバッグ

AUTOSAR Debugging によってアクセスされる Watchdog Manager の各変数をグローバル変数として定義するものとする【WDGM234】。

AUTOSAR Debugging によってアクセスされる Watchdog Manager の変数の全ての型は WdgM.h によってアクセス可能とするものとする【WDGM235】。

AUTOSAR Debugging によってアクセスされる WdgM.h 内の Watchdog Manager の変数の宣言は、C 言語の sizeof 演算子によって変数のサイズを計算できるようにするものとする【WDGM236】。

AUTOSAR Debugging によってアクセスされる Watchdog Manager の変数はそれぞれの BSWM 記述内で説明されること。AUTOSAR Debugging によってアクセスされる変数を以下に示す【WDGM237】。

- ・各監視エンティティのローカル監視状態【WDGM238】
- ・グローバル監視状態【WDGM239】
- ・各チェックポイントのアライブカウンタ【WDGM240】
- ・チェックポイントに到達した時のタイムスタンプ【WDGM241】
- ・到達したチェックポイント【WDGM242】

3.2 データ型

3.2.1 外部からインポートするデータ型

Watchdog Manager は, 表 3-1 に示すデータ型をインポートして使用するものとする【WDGM011】.

表 3-1 インポートするデータ型

ファイル名	データ型
Dem	Dem_EventIdType
	Dem_EventStatusType
Os	AccessType
	ApplicationStateRefType
	ApplicationType
	CounterType
	ISRTType
	MemorySizeType
	MemoryStartAddressType
	ObjectAccessType
	ObjectTypeType
	RestartType
	ScheduleTableStatusRefType
	ScheduleTableType
	StatusType
	TaskType
	TickRefType
	TickType
Std_Types	TrustedFunctionIndexType
	TrustedFunctionParameterRefType
Std_Types	Std_ReturnType
	Std_VersionInfoType
WdgIf	WdgIf_ModeType

3.2.2 内部で定義するデータ型

Watchdog Manager は、以下で示すデータ型を使用するものとする【WDGM038】.

3.2.2.1 WdgM_ConfigType

データ型名	WdgM_ConfigType
型	構造体型
値の範囲	実装仕様
概要	この構造体は Watchdog Manager のポストビルドコンフィギュレーションパラメータを全て含む. コンフィギュレーションのため, このポインタが WdgM_Init に渡される.

WdgM_ConfigType は Watchdog Manager のポストビルドコンフィギュレーションパラメータを全て含むものとする. この構造体の正確なメンバは, 選択されたコンフィギュレーションバリエーションに依存する. コンフィギュレーションパラメータについては, 3.10 コンテナおよびパラメータを参照のこと【WDGM042】.

3.2.2.2 WdgM_SupervisedEntityIdType

データ型名	WdgM_SupervisedEntityIdType
型	uint8, uint16
値の範囲	0..< 監視エンティティ ID の最大値 > 0 から連番となる整数値
概要	監視エンティティ ID 型

3.2.2.3 WdgM_CheckpointIdType

データ型名	WdgM_CheckpointIdType
型	uint8, uint16
値の範囲	0..< チェックポイント ID の最大値 > 0 から連番となる整数値
概要	チェックポイント ID 型

チェックポイント ID は単体では固有の値ではないため注意すること. 監視エンティティ ID とチェックポイント ID の 2 つを併用することで, チェックポイントを識別できる.

3.2.2.4 WdgM_ModeType

データ型名	WdgM_ModeType	
型	uint8	
値の範囲	0..< Watchdog Manager モードの最大値 >	0 から連番となる整数値
概要	Watchdog Manager モード型	

3.2.2.5 WdgM_LocalStatusType

データ型名	WdgM_LocalStatusType	
型	uint8	
値の範囲	WDGM_LOCAL_STATUS_OK	0
	WDGM_LOCAL_STATUS_FAILED	1
	WDGM_LOCAL_STATUS_EXPIRED	2
	WDGM_LOCAL_STATUS_DEACTIVATED	4
概要	監視エンティティ個別のローカル監視状態型	

3.2.2.6 WdgM_GlobalStatusType

データ型名	WdgM_GlobalStatusType	
型	uint8	
値の範囲	WDGM_GLOBAL_STATUS_OK	0
	WDGM_GLOBAL_STATUS_FAILED	1
	WDGM_GLOBAL_STATUS_EXPIRED	2
	WDGM_GLOBAL_STATUS_STOPPED	3
	WDGM_GLOBAL_STATUS_DEACTIVATED	4
概要	Watchdog Manager のグローバル監視状態型	

3.3 定数

3.3.1 WdgM_LocalStatusType

WDGM_LOCAL_STATUS_OK

定数名	WDGM_LOCAL_STATUS_OK
概要	この監視エンティティの監視結果は異常がない。

WDGM_LOCAL_STATUS_FAILED

定数名	WDGM_LOCAL_STATUS_FAILED
概要	この監視エンティティの監視結果は異常があるが復帰する可能性がある。すなわち、監視エンティティが通常ふるまいに戻った場合、監視状態もWDGM_LOCAL_STATUS_OKに戻る。そのうえ、監視エンティティが異常になってからの回数が構成された許容回数を上回っていない。この許容回数を上回った時、監視状態がWDGM_LOCAL_STATUS_EXPIREDになる。

WDGM_LOCAL_STATUS_EXPIRED

定数名	WDGM_LOCAL_STATUS_EXPIRED
概要	この監視エンティティの監視結果が恒久的に異常となった。この状態から変わることはない。

WDGM_LOCAL_STATUS_DEACTIVATED

定数名	WDGM_LOCAL_STATUS_DEACTIVATED
概要	この監視エンティティの監視が一時的に停止している。

3.3.2 WdgM_GlobalStatusType

WDGM_GLOBAL_STATUS_OK

定数名	WDGM_GLOBAL_STATUS_OK
概要	監視に異常がない。

WDGM_GLOBAL_STATUS_FAILED

定数名	WDGM_GLOBAL_STATUS_FAILED
概要	監視に異常があるが、許容範囲内である。

WDGM_GLOBAL_STATUS_EXPIRED

定数名	WDGM_GLOBAL_STATUS_EXPIRED
概要	監視に異常がありかつ異常の許容数の限界を上回っているが、Watchdog Driver はまだトリガ操作を停止する指示をされていない。

WDGM_GLOBAL_STATUS_STOPPED

定数名	WDGM_GLOBAL_STATUS_STOPPED
概要	監視に異常がありかつ異常の許容数の限界を上回っており、Watchdog Driver がトリガ操作の停止指示をされている。結果として Watchdog リセットが発生する。

WDGM_GLOBAL_STATUS_DEACTIVATED

定数名	WDGM_GLOBAL_STATUS_DEACTIVATED
概要	Watchdog Manager が初期化されていないかつ Watchdog の管理がされていない。

3.4 返り値

Watchdog Manager はシステムサービスの返り値のためのデータ型を定義しない。Watchdog Manager システムサービスは Std_ReturnType の値を返す。

Watchdog Manager システムサービスの返り値を以下の表 3-2 に示す。

表 3-2 Watchdog Manager システムサービスの返り値一覧

返り値名	値	ファイル名	概要
E_OK	0x00	Std_Types.h	システムサービス正常終了
E_NOT_OK	0x01	Std_Types.h	システムサービス異常終了

3.5 エラーコード

拡張エラーは uint8 型で定義すること【WDGM259】。

Watchdog Manager はコンフィギュレーション(development mode または production mode)に基づき、以下の表 3-4 で示す標準/拡張エラーを検出可能であるものとする。標準エラーコードのイベント ID 値は DEM により外部で割り当てられる【WDGM004】。

3.5.1 標準エラー

表 3-3 標準エラーコード一覧

エラーコード	値	概要
WDGM_E_SUPERVISION	DEM で定義	監視に異常があり、かつ Watchdog のリセットが発生しようとしている。
WDGM_E_SET_MODE	DEM で定義	Watchdog Driver のモード切替が失敗した。
WDGM_E_IMPROPER_CALLER	DEM で定義	Defensive behavior check が不正なコールを検出した。
WDGM_E_MFCYCLE_OR_OSCOUNTER	DEM で定義	WdgM_MainFunction のコール周期と OS カウンタから取得された経過時間が乖離した。
WDGM_E_OSCOUNTER	DEM で定義	OS カウンタからのタイムスタンプの取得に失敗した。

3.5.2 拡張エラー

表 3-4 拡張エラーコード一覧

エラーコード	値	概要
WDGM_E_NO_INIT	0x10	API サービスがモジュールの初期化前に使用された。
WDGM_E_PARAM_CONFIG	0x11	API サービス WdgM_Init が間違ったコンフィギュレーションセットでコールされた。
WDGM_E_PARAM_MODE	0x12	API サービスが間違った mode パラメータでコールされた。
WDGM_E_PARAM_SEID	0x13	API サービスが間違った supervised entity identifier パラメータでコールされた。
WDGM_E_INV_POINTER	0x14	NULL ポインタのパラメータで API サービスがコールされた。
WDGM_E_DISABLE_NOT_ALLOWED	0x15	Watchdog の無効化が認められていない。
WDGM_E_CPID	0x16	API サービスが無効な CheckpointId で使用された。
WDGM_E_DEPRECATED	0x17	咎められた API サービスが使用された。
WDGM_E_AMBIGIOUS	0x18	与えられた監視エンティティの現在モードで一つより多くのアライブ監視が存在していたため、 WdgM_UpdateAliveIndication がチェックポイントを決定できなかった。
WDGM_E_SEDEACTIVATED	0x19	API サービスが現在の Watchdog Manager mode で非アクティブな監視エンティティのチェックポイントで使用された。
WDGM_E_CONCURRENTLY_CALLED	0xA0	同時動作不可能なシステムサービスが同時にコールされた。

3.6 API

3.6.1 WdgM_Init

サービス ID		0x00	
C 言語 I/F		void WdgM_Init(const WdgM_ConfigType* ConfigPtr) 【WDGM151】	
パラメータ	[in]	ConfigPtr	コンフィギュレーション情報へのポインタ
	[in/out]	—	
	[out]	—	
返り値		—	
DET エラー		WDGM_E_INV_POINTER	引数に NULL ポインタが指定された 【WDGM255】
		WDGM_E_PARAM_CONFIG	不正なコンフィギュレーションパラメータが与えられた 【WDGM010】
同期/非同期		同期	
再入可能/再入不可能		再入不可能	
機能		本 API は、Watchdog Manager の初期化を行う。	

機能仕様

この機能は Watchdog Manager の初期化をする。この機能の実行後、監視が初期モードで定義された監視エンティティのリストに従ってアクティブ化される。

本 API は、Watchdog Manager の全ての変数を初期化するものとする 【WDGM018】。

本 API は、Watchdog Manager の初期モードを確立するものとする。初期モードが確立できなかった場合のふるまいは 【WDGM139】 参照のこと。WdgMDevErrorDetect が有効の場合にだけ実行するオプションのチェックが存在する 【WDGM135】。

WdgMDevErrorDetect が有効かつコンフィギュレーションバリエーションが VARIANT-POST-BUILD の場合、本 API は ConfigPtr が NULL ポインタを指していないかチェックするものとする。ConfigPtr が NULL ポインタを指していた場合、本 API は処理をしないものとする。また、DET のサービス Det_ReportError に WDGM_E_INV_POINTER をレポートするものとする 【WDGM255】。

WdgMDevErrorDetect 【WDGM301_Conf】 が有効かつコンフィギュレーションバリエーションが VARIANT-POST-BUILD の場合、コンフィギュレーションセットにより与えられるメンバが許可され

ている境界内に入っているかチェックするものとする。エラーを検出した場合、本 API は初期化処理をしないものとする。また、DET のサービス `Det_ReportError` に `WDGM_E_PARAM_CONFIG` をレポートするものとする【WDGM010】。

最初に EXPIRED となった SEID と SEID の反転値を non-initialized RAM からクリアするものとする【WDGM350】。

3.6.2 WdgM_DeInit

サービス ID		0x01
C 言語 I/F		void WdgM_DeInit(void) 【WDGM261】
パラメータ	[in]	—
	[in/out]	—
	[out]	—
返り値		—
DET エラー	WDGM_E_NO_INIT	初期化未実施 【WDGM288】
	WDGM_E_CONCURRENTLY_CALLED	同時動作不可能なシステムサービスが同時にコールされた 【NWDGM102】
同期/非同期		同期
再入可能/再入不可能		再入不可能
機能		本 API は、Watchdog Manager を非初期化状態にする。

機能仕様

WdgMDevErrorDetect が有効の場合、オプションのチェックが存在する。

WdgMDevErrorDetect が有効の場合、本 API は Watchdog Manager が初期化されているかをチェックするものとする。初期化されていない場合、本 API は処理をしないものとする。また、DET のサービス Det_ReportError に WDGM_E_NO_INIT をレポートするものとする 【WDGM288】。

WdgM_PerformReset がコールされていた場合、WdgM_DeInit はなにもしない 【NWDGM070】。

WdgM_DeInit は WdgM_SetMode、WdgM_MainFunction と同時にコールされてはならない。WdgMDevErrorDetect が有効の場合他のシステムサービスが動作中かどうかを確認し、もし同時にコールされれば、DET のサービス Det_ReportError に WDGM_E_CONCURRENTLY_CALLED をレポートするものとする 【NWDGM102】。

使用上の注意

このサービスはウィンドウ Watchdog のコンセプトの結果として必要である。

Watchdog Manager が停止する前に、Watchdog リセットが起こることなく、残りのシャットダウン処理が実行されるようなタイムアウト値となるように、全ての起動中の Watchdog のトリガコンディションをセットするものとする。

停止処理用のモードをコンフィグレーションし、WdgM_DeInit の前に WdgM_SetMode によってそ

のモードに切り替えることを推奨する。

3.6.3 WdgM_GetVersionInfo

サービス ID		0x02	
C 言語 I/F		void WdgM_GetVersionInfo(Std_VersionInfoType* VersionInfo) 【WDGM153】	
パラメータ	[in]	—	
	[in/out]	—	
	[out]	VersionInfo	バージョン情報格納先へのポインタ
返り値		—	
DET エラー		WDGM_E_INV_POINTER	引数に NULL ポインタが指定された 【WDGM256】
同期/非同期		同期	
再入可能/再入不可能		再入可能	
機能		Watchdog Manager のバージョン情報を返す。	

機能仕様

本 API は Watchdog Manager のバージョン情報を返すものとする。バージョン情報はモジュール ID、ベンダーID、ベンダー固有のバージョン番号を含むものとする。本 API はプレコンパイル時にパラメータ WdgMVersionInfoApi 【WDGM302_Conf】により無効化することができる【WDGM110】。

WdgMDevErrorDetect が有効の場合、オプションのチェックが存在する

WdgMDevErrorDetect 【WDGM301_Conf】が有効の場合、本 API は VersionInfo が NULL ポインタを指していないかチェックするものとする。VersionInfo が NULL ポインタを指していた場合、本 API は処理しないものとする。また、DET のサービス Det_ReportError に WDGM_E_INV_POINTER をレポートするものとする【WDGM256】。

設計上の注意

本 API のコール元およびコール先のソースコードがコンパイル時に入手される場合は、モジュールのヘッダーファイルの中で定義して、WdgM_GetVersionInfo をマクロとして実現すべきである。【WDGM262】。

3.6.4 WdgM_SetMode

サービス ID		0x03	
C 言語 I/F		Std_ReturnType WdgM_SetMode(WdgM_ModeType Mode, uint16 CallerID) 【WDGM154】	
パラメータ	[in]	Mode	Watchdog Manager のコンフィギュレーションモード
		CallerID	コール元のモジュール ID
	[in/out]	－	
	[out]	－	
返り値		E_OK	新しいモードへの切替成功
		E_NOT_OK	新しいモードへの切替失敗
DET エラー		WDGM_E_SET_MODE	定義されていない原因のエラー 【WDGM142】
		WDGM_E_PARAM_MODE	不正な入力パラメータが与えられた 【WDGM020】
		WDGM_E_NO_INIT	初期化未実施【WDGM021】
		WDGM_E_IMPROPER_CALLER	未定義の CallerID が入力パラメータとして与えられた【NWDGM046】
		WDGM_E_CONCURRENTLY_CALLED	同時動作不可能なシステムサービスが同時にコールされた【NWDGM103】
同期/非同期		同期	
再入可能/再入不可能		再入可能	
機能		Watchdog Manager の現在のモードを設定する。	

機能仕様

Watchdog Manager は、グローバル監視状態が WDGM_GLOBAL_STATUS_OK または WDGM_GLOBAL_STATUS_FAILED の場合だけ、本 API を実行するものとする【WDGM145】。

本 API 内で WdgIf_SetMode が失敗した場合、Watchdog Manager は DEM に WDGM_E_SET_MODE としてレポートするものとする【NWDGM042】。

WdgMDevErrorDetect が有効の場合、オプションのチェックが存在する【WDGM142_2】。

WdgMDevErrorDetect が有効の場合、入力パラメータの Mode が許容範囲内であるかチェックするものとする。エラーの場合、モード切替は実行されないものとする。また、DET に WDG_M_E_PARAM_MODE としてレポートすること、および戻り値として E_NOT_OK を返すものとする【WDGM020】。

WdgMDevErrorDetect が有効の場合、Watchdog Manager が初期化されているかチェックするものとする。エラーの場合、モード切替は実行されないものとする。また、DET に WDG_M_E_NO_INIT としてレポートすること、および戻り値として E_NOT_OK を返すものとする【WDGM021】。

WdgMDefensiveBehavior が有効の場合、Watchdog Manager は入力パラメータとして与えられた CallerID が許可されている CallerID のリストの中に存在するかチェックするものとする。もし存在しない CallerID であった場合、影響を及ぼすことなく戻り値として E_NOT_OK を返すものとする。また、DEM に WDG_M_E_IMPROPER_CALLER をレポートするものとする【NWDGM046】。

WdgM_PerformReset がコールされていた場合、WdgM_SetMode はなにもしない【NWDGM071】。

WdgM_SetMode は WdgM_DeInit, WdgM_MainFunction と同時にコールされてはならない。WdgMDevErrorDetect が有効の場合は他のシステムサービスが動作中かどうかを確認し、もし同時にコールされれば、DET のサービス Det_ReportError に WDG_M_E_CONCURRENTLY_CALLED をレポートするものとする【NWDGM103】。

3.6.5 WdgM_GetMode

サービス ID		0x0b	
C 言語 I/F		Std_ReturnType WdgM_GetMode(WdgM_ModeType* Mode) 【WDGM168】	
パラメータ	[in]	—	
	[in/out]	—	
	[out]	Mode	Watchdog Manager の現在のモード
返り値	E_OK		現在のモードを返すことに成功
	E_NOT_OK		現在のモードを返すことに失敗
DET エラー	WDGM_E_NO_INIT		初期化未実施 【WDGM253】
	WDGM_E_INV_POINTER		引数に NULL ポインタが指定された 【WDGM254】
同期/非同期		同期	
再入可能/再入不可能		再入可能	
機能		Watchdog Manager の現在のモードを返す	

機能仕様

本 API は Watchdog Manager の現在のアクティブモードを返すものとする。この API がコールされている間に WdgM_SetMode がアクティブとなった場合、新しいモードが完全にアクティブになるまでは、前回のアクティブモードを返すものとする 【WDGM170】。

WdgMDevErrorDetect が有効の場合のみ、実施されるオプションのチェックが存在する。

WdgMDevErrorDetect が有効の場合、本 API は Watchdog Manager が初期化されているかチェックするものとする。初期化されていない場合、処理を実行してはならない。また、DET に WDGM_E_NO_INIT をレポートして、返り値として E_NOT_OK を返すものとする 【WDGM253】。

WdgMDevErrorDetect が有効の場合、出力パラメータが NULL ポインタであるかチェックするものとする。エラーの場合には、処理を実行しないものとする。また、DET に WDGM_E_INV_POINTER をレポートして、返り値として E_NOT_OK を返すものとする 【WDGM254】。

3.6.6 WdgM_CheckpointReached

サービス ID		0x0e	
C 言語 I/F		Std_ReturnType WdgM_CheckpointReached(WdgM_SupervisedEntityIdType SEID, WdgM_CheckpointIdType CheckpointID) 【WDGM263】	
パラメータ	[in]	SEID	チェックポイントを報告する監視エンティティの ID
		CheckpointID	監視エンティティ内の到達したチェックポイントの ID
	[in/out]	—	
	[out]	—	
返り値		E_OK	アライブカウンタのアップデート成功
		E_NOT_OK	アップデート失敗
DET エラー		WDGM_E_PARAM_SEID	引数に管理外の SEID が指定された 【WDGM278】
		WDGM_E_NO_INIT	初期化未実施 【WDGM279】
		WDGM_E_CPID	引数に管理外のチェックポイントが指定された 【WDGM284】
		WDGM_E_SEDEACTIVATED	非アクティブな監視エンティティが指定された 【WDGM319】
同期/非同期		同期	
再入可能/再入不可能		再入可能	
機能		監視エンティティ内のチェックポイントに到達したことを Watchdog Manager に通知	

機能仕様

本 API は、レポートされたチェックポイントのアライブ通知カウンタをインクリメントするものとする【WDGM321】。

本 API は、レポートされたチェックポイントを使用する監視エンティティのデッドライン監視するものとする。出力は監視エンティティのデッドライン監視の更新された結果であるものとする【WDGM322】。

本 API は、レポートされたチェックポイントを使用する監視エンティティのロジカル監視をするものとする。出力は監視エンティティのロジカル監視の更新結果であるものとする【WDGM323】。

WdgMDevErrorDetect が有効の場合のみ、実行されるオプションのチェックが存在する。

コンフィギュレーションパラメータ WdgMDevErrorDetect [WDGM301_Conf] が有効の場合、パラメータ SEID は Watchdog Manager のコントロール下のエンティティのリスト内かチェックされるものとする。エラーの場合、処理が実行しないものとする。また、DET に WDGM_E_PARAM_SEID としてレポートおよび、戻り値として E_NOT_OK を返すものとする【WDGM278】。

コンフィギュレーションパラメータ WdgMDevErrorDetect [WDGM301_Conf] が有効の場合、Watchdog Manager が初期化されているかチェックするものとする。エラーの場合には、処理が実行されないものとする。また、DET に WDGM_E_NO_INIT としてレポートおよび、戻り値として E_NOT_OK を返すものとする【WDGM279】。

コンフィギュレーションパラメータ WdgMDevErrorDetect [WDGM301_Conf] が有効の場合、パラメータ CheckpointID が、パラメータ SEID によって与えられる監視エンティティと関連したチェックポイントのセット [WDGM_Conf303] 内であるかどうかチェックするものとする。エラーの場合、処理が実行されないものとする。また、DET に WDGM_E_CPID としてレポートおよび、戻り値として E_NOT_OK を返すものとする【WDGM284】。

コンフィギュレーションパラメータ WdgMDevErrorDetect [WDGM301_Conf] が有効の場合、CheckpointID が所属している監視エンティティがアクティブかどうかチェックするものとする。監視エンティティがアクティブでない場合、処理が実行されないものとする。また、DET に WDGM_E_SEDEACTIVATED としてレポートおよび、戻り値として E_NOT_OK を返すものとする【WDGM319】。

3.6.7 WdgM_GetLocalStatus

サービス ID		0x0c	
C 言語 I/F		Std_ReturnType WdgM_GetLocalStatus (WdgM_SupervisionEntityIdType SEID, WdgM_LocalStatusType* Status) 【WDGM169】	
パラメータ	[in]	SEID	監視状態が返される監視エンティティ ID
	[in/out]	—	
	[out]	Status	与えられた監視エンティティの監視状態
返り値	E_OK		監視状態を返すことに成功
	E_NOT_OK		監視状態を返すことに失敗
DET エラー	WDGM_E_PARAM_SEID		引数に管理外の SEID が指定された 【WDGM172】
	WDGM_E_INV_POINTER		出力に NULL ポインタが指定された 【WDGM257】
	WDGM_E_NO_INIT		初期化未実施 【WDGM173】
同期/非同期		同期	
再入可能/再入不可能		再入可能	
機能		監視エンティティ個別の監視状態を返す	

機能仕様

WdgM_GetLocalStatus は与えられた監視エンティティの個別の監視状態を返すものとする
 【WDGM171】.

WdgMDevErrorDetect が有効の場合のみ、実施されるオプションのチェックが存在する.

コンフィギュレーションパラメータ WdgMDevErrorDetect 【WDGM301_Conf】 が有効の場合、パラメータ SEID は Watchdog Manager のコントロール下のエンティティのリスト内かチェックするものとする. エラーの場合、処理が実行されないものとする. また、DET に WDGM_E_PARAM_SEID としてレポートおよび、返り値として E_NOT_OK を返すものとする 【WDGM172】.

コンフィギュレーションパラメータ WdgMDevErrorDetect 【WDGM301_Conf】 が有効の場合、出力パラメータが NULL ポインタを指していないかチェックするものとする. エラーの場合、処理が実行されないものとする. また、DET に WDGM_E_INV_POINTER としてレポートおよび、返り値として E_NOT_OK を返すものとする 【WDGM257】.

コンフィギュレーションパラメータ WdgMDevErrorDetect 【WDGM301_Conf】 が有効の場合、

Watchdog Manager が初期化されているかチェックするものとする。エラーの場合、処理が実行されないものとする。また、DET に WDGGM_E_NO_INIT としてレポートおよび、返り値として E_NOT_OK を返すものとする【WDGM173】。

3.6.8 WdgM_GetGlobalStatus

サービス ID		0x0d	
C 言語 I/F		Std_ReturnType WdgM_GetGlobalStatus (WdgM_GlobalStatusType* Status) 【WDGM175】	
パラメータ	[in]	—	
	[in/out]	—	
	[out]	Status	WdgM のグローバル監視状態
返り値	E_OK		監視状態を返すことに成功
	E_NOT_OK		監視状態を返すことに失敗
DET エラー	WDGM_E_INV_POINTER		出力に NULL ポインタが指定された 【WDGM258】
同期/非同期	同期		
再入可能/再入不可能	再入可能		
機能	WdgM のグローバル監視状態を返す。		

機能仕様

WdgM の開発エラー検出が有効の場合、パラメータ Status が NULL ポインタを指していないかチェックするものとする。Status が NULL ポインタの場合、開発エラーWDGM_E_INV_POINTER を発生させて返すものとする 【WDGM344】。

WdgMDevErrorDetect が有効の場合のみ、実施されるオプションのチェックが存在する。

コンフィギュレーションパラメータ WdgMDevErrorDetect 【WDGM301_Conf】が有効の場合、出力パラメータが NULL ポインタを指していないかチェックするものとする。エラーの場合、処理が実行されないものとする。また、DET に WDGM_E_INV_POINTER としてレポートおよび、返り値として E_NOT_OK を返すものとする 【WDGM258】。

この機能は、WdgM_Init より前に利用できるものとする 【NWDGM059】。

3.6.9 WdgM_PerformReset

サービス ID		0x0f
C 言語 I/F		void WdgM_PerformReset (void) 【WDGM264】
パラメータ	[in]	—
	[in/out]	—
	[out]	—
返り値		—
DET エラー	WDGM_E_NO_INIT	初期化未実施 【WDGM270】
同期/非同期	同期	
再入可能/再入不可能	再入不可能	
機能	Watchdog Manager に Watchdog リセットを引き起こすように指示する。	

機能仕様

このサービスがコールされた時、Watchdog Manager は全ての構成された Watchdog Driver のトリガコンディションを 0 にセットするものとする 【WDGM232】。

それによって、Watchdog タイマは外部ハードウェアのリセットを引き起こす。

このサービスがコールされた後、Watchdog Manager はトリガコンディションを更新しないものとする 【WDGM233】。

このシステムサービスがコールされた時、グローバル監視状態はもはや考慮されない。

WdgMDevErrorDetect が有効の場合のみ、実施されるオプションのチェックが存在する。

コンフィギュレーションパラメータ WdgMDevErrorDetect 【WDGM301_Conf】 が有効の場合、Watchdog Manager が初期化されているかチェックするものとする。エラーの場合、処理が実行されないものとする。また、DET に WDGM_E_NO_INIT としてレポートするものとする 【WDGM270】。

WdgM_PerformReset がコールされていた場合、WdgM_PerformReset はなにもしない 【NWDGM073】。

3.6.10 WdgM_GetFirstExpiredSEID

サービス ID		0x10	
C 言語 I/F		Std_ReturnType WdgM_GetFirstExpiredSEID(WdgM_SupervisedEntityIdType* SEID) 【WDGM346】	
パラメータ	[in]	—	
	[in/out]	—	
	[out]	SEID	最初に WDGM_LOCAL_STATUS_EXPIRED の状態となった監視エンティティの ID
返り値	E_OK		SEID を返すことに成功
	E_NOT_OK		SEID を返すことに失敗
DET エラー	WDGM_E_INV_POINTER		引数に NULL ポインタが指定された 【WDGM347】
同期/非同期	同期		
再入可能/再入不可能	再入可能		
機能	最初に WDGM_LOCAL_STATUS_EXPIRED の状態となった SEID を返す。		

機能仕様

Watchdog Manager の開発エラー検出が有効の場合、パラメータ SEID が NULL ポインタかどうかをチェックするものとする。SEID が NULL ポインタである場合は、DET に WDGM_E_INV_POINTER としてレポートおよび、返り値として E_NOT_OK を返すものとする 【WDGM347】。

この機能は、WdgM_Init より前に利用できるものとする 【WDGM348】。

この機能は、non-initialized RAM から SEID と SEID の反転値を読むものとする。値と反転値が互いに一致しなかった場合、E_NOT_OK を返し、SEID に 0 を書き込むものとする。値と反転値が一致した場合、E_OK を返し、SEID に読んだ値を書き込むものとする 【WDGM349】。

3.7 コールバック関数

Watchdog Manager はコールバック関数を提供しない。

3.8 周期関数

3.8.1 WdgM_MainFunction

サービス ID	0x08	
C 言語 I/F	void WdgM_MainFunction(void) 【WDGM159】	
起動タイミング	FIXED_CYCLIC	
DET エラー	WDGM_E_NO_INIT	初期化未実施 [NWDGM104]
	WDGM_E_CONCURRENTLY_CALLED	同時動作不可能なシステムサービスが同時にコールされた [NWDGM104]
機能	WdgM の周期的な責務の処理を行う。	

機能仕様

WdgM_MainFunction は Watchdog Manager の周期的な責務の処理を行うものとする【WDGM060】。

WdgM_MainFunction は報告されたチェックポイントを使用する報告された監視エンティティのアライブ監視を行うものとする。この機能の入力はチェックポイントのアライブ通知カウンタであるものとする。この機能の出力は監視エンティティのアライブ監視の結果であるものとする【WDGM324】。

アライブ、デッドラインやロジカル監視からの結果に基づき、アクティブな各監視エンティティのために、WdgM_MainFunction はローカル監視状態を決定するものとする【WDGM325】。

WdgM_Init が最後にコールされてから、最初に WDGM_LOCAL_STATUS_EXPIRED に切り替わった監視エンティティのために、WdgM_MainFunction は初期化されない non-initialized RAM に監視エンティティの SEID と SEID の反転値を格納するものとする【WDGM351】。

アクティブな各監視エンティティのローカル監視状態に基づき、WdgM_MainFunction は、グローバル監視状態を決定するものとする【WDGM326】。

各監視エンティティのローカル監視状態とグローバル監視状態に基づき、WdgM_MainFunction は付随するエラーハンドリングを管理するものとする【WDGM327】。

グローバル監視状態に基づき、WdgM_MainFunction は Watchdog Interface の付随するトリガコンディション設定をコールするものとする【WDGM328】。

グローバル監視状態が `WDGM_GLOBAL_STATUS_DEACTIVATED` でない場合、`WdgM_MainFunction` は構成された監視サイクル (`WdgMSupervisionCycle` [`WDGM330_Conf`] 参照) に従い実行されるものとする【WDGM063_1】。

監視エンティティがデッドロックで終了して脱出しない場合、Watchdog Manager の main function は、コールされることがなく、フェイルとなった監視エンティティを検出しない。従って、main function を含むタスクは Watchdog Manager によって監視される監視エンティティを含むタスクとは分けるものとする【WDGM063_2】。

`WdgM_MainFunction` を実行する OS タスクは、監視下の監視エンティティからコールされる機能をコールする OS タスクとは分けるものとする【WDGM275】。

コンフィギュレーションパラメータ `WdgMDevErrorDetect` が有効の場合、Watchdog Manager が初期化されているか確認するものとする。エラーの場合、`WdgM_MainFunction` は実行されないものとする。また、DET にエラーコード `WDGM_E_NO_INIT` を報告するものとする【NWDGM047】。

`WdgM_PerformReset` がコールされていた場合、`WdgM_MainFunction` はなにもしない【NWDGM074】。

`WdgM_MainFunction` は `WdgM_SetMode`、`WdgM_DeInit` と同時にコールされてはならない。`WdgMDevErrorDetect` が有効の場合は他のシステムサービスが動作中かどうかを確認し、もし同時にコールされれば、DET のサービス `Det_ReportError` に `WDGM_E_CONCURRENTLY_CALLED` をレポートするものとする【NWDGM104】。

3.9 依存インタフェース

3.9.1 必須インタフェース

Watchdog Manager の機能を実現するために、表 3-5 で示すインタフェースが必要となる【WDGM161】.

表 3-5 必須インタフェース

システムサービス名	概要
AllowAccess	OS アプリケーション自身の状態を APPLICATION_RESTARTING から APPLICATION_ACCESSIBLE に設定するサービス.
CallTrustedFunction	信頼機能をコールするために使用されるサービス.
CheckISRMemoryAccess	メモリ領域が write/read/execute accessible であるかどうかの確認とメモリ領域がスタック領域の一部であるかどうかの情報を返すサービス.
CheckObjectAccess	ApplID によって与えられた OS アプリケーションがタスクの ID, ISR, リソース, カウンタ, アラームやスケジュールテーブルを API コールの中で使用することが許可されているかどうかを決定するサービス.
CheckObjectOwnership	与えられたタスク, ISR, カウンタ, アラームやスケジュールテーブルが OS アプリケーションに所属しているかを決定する.
CheckTaskMemoryAccess	メモリ領域が write/read/execute accessible であるかどうかの確認とメモリ領域がスタック領域の一部であるかどうかの情報を返すサービス.
GetApplicationID	現在実行中の OS アプリケーションを決定するサービス.
GetApplicationState	OS アプリケーションの現在の状態を返すサービス.
GetCounterValue 【NWDGM087】	現在の tick 値を取得するサービス.
GetElapsedValue	現在の tick 値と前回読んだ tick 値の差分の tick 値を取得するサービス.
GetISRID	現在実行中の ISR の ID を返すサービス.
GetScheduleTableStatus	スケジュールテーブルの状態を問い合わせるサービス.
IncrementCounter	ソフトウェアカウンタをインクリメントするサービス.
NextScheduleTable	あるスケジュールテーブルから他のスケジュールテーブルへ処理を切り替えサービス.
SetScheduletableAsync	スケジュールテーブルの同期を停止するサービス.
StartScheduleTableAbs	下層のカウンタの絶対値である開始値でスケジュールテーブルの処理を開始するサービス.

StartScheduleTableRel	下層のカウンタの現在値に関連したオフセットでスケジュールテーブルの処理を開始するサービス.
StartScheduleTableSynchron	明確に同期されたスケジュールテーブルを同期して開始させるサービス.
StopScheduleTable	スケジュール実行中の如何なるポイントでもすぐにスケジュールテーブルの処理を中止する.
SyncScheduleTable	スケジュールテーブルに同期カウントを提供し, 同期を開始するサービス
WdgIf_SetMode	一致する Watchdog Driver の Wdg_SetMode に割り当てるサービス.
WdgIf_SetTriggerCondition	一致する Watchdog Driver の Wdg_SetTriggerCondition を割り当てるサービス.

3.9.2 任意インタフェース

Watchdog Manager の任意の機能を実現するために、表 3-6 で示すインタフェースが必要となる【WDGM162】.

表 3-6 任意インタフェース

システムサービス名	概要
BswM_WdgM_RequestPartitionReset	パーティションのリセットを要求するために WdgM からコールされるサービス.
Dem_ReportErrorStatus	BSW モジュールから報告されたイベントをキューに入れるサービス. イベントの処理は Dem の main function 内で行われるため、非同期のふるまいをする.
Det_ReportError	開発エラーを報告するためのサービス.
Mcu_PerformReset	マイコンのリセットをするサービス.
TerminateApplication	エラーフックがコールされたタスク/C2ISR/アプリケーションが所属する OS アプリケーションを終了するサービス.

DEM エラー発生時にシステムサービス Dem_ReportErrorStatus を使用して、エラー報告を行う必要があるが、引数で必要となる EventStatus が Watchdog Manager の仕様内で指定されておらず、DEM の参照コンテナ内にも設定箇所が見られない. Watchdog Manager が DEM を発行する条件は全て正常に動作していない場合であることから DEM_EVENT_STATUS_FAILED を指定する【NWDGM043】.

3.9.3 コンフィギュレーション可能なインタフェース

Watchdog Manager はコンフィギュレーション可能なインタフェースを持たない.

3.10 コンテナおよびパラメータ

3.10.1 WdgM

コンテナ名	/AUTOSAR/EcucDefs/WdgM 【WDGM_Conf001】
概要	WdgM トップコンテナ
多重度	0..1
パラメータ	-
サブコンテナ	WdgMConfigSet
	WdgMGeneral

3.10.2 WdgMGeneral

コンテナ名	/AUTOSAR/EcucDefs/WdgM/WdgMGeneral 【WDGM_Conf300】
概要	全般的な設定を含むコンテナ
多重度	1
パラメータ	WdgMDefensiveBehavior
	WdgMDemStoppedSupervisionReport
	WdgMDevErrorDetect
	WdgMImmediateReset
	WdgMOffModeEnabled
	WdgMDeadlineTimeoutEnabled [NWDGM_Conf077]
	WdgMMMFCycleAndOsCounterSupervisionEnabled [NWDGM_Conf082]
	WdgMVersionInfoApi
サブコンテナ	WdgMCallerIds
	WdgMSupervisedEntity
	WdgMWatchdog

3.10.2.1 WdgMDefensiveBehavior

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMDefensiveBehavior 【WDGM_Conf352】	
概要	防御行動の有効/無効切り替え〔NWDGM046〕	
多重度	1	
型	EcucBooleanParamDef	
値の範囲	TRUE	DET 有効
	FALSE	DET 無効
デフォルト値	FALSE	
コンフィギュレーションクラス		
プレコンパイルタイム		

本コンフィグパラメータの値を WDGM_DEFENSIVE_BEHAVIOR として参照可能とすること【NWDGM116】.

3.10.2.2 WdgMDemStoppedSupervisionReport

パラメータ名	/AUTOSAR/EcucDefs/WdgM/WdgMGeneral/ WdgMDemStoppedSupervisionReport 【WDGM_Conf338】	
概要	DEM への WDGM_GLOBAL_STATUS_STOPPED への到達エラーレポート有効有無	
多重度	1	
型	EcucBooleanParamDef	
値の範囲	TRUE	DEM へのエラーレポート有効
	FALSE	DEM へのエラーレポート無効
デフォルト値	－	
コンフィギュレーションクラス		
プレコンパイルタイム		

本コンフィグパラメータの値を WDGM_DEM_STOPPED_SUPERVISION_REPORT として参照可能とすること【NWDGM117】.

3.10.2.3 WdgMDevErrorDetect

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMDevErrorDetect 【WDGM_Conf301】	
概要	DET の有効有無	
多重度	1	
型	EcucBooleanParamDef	
値の範囲	TRUE	DET 有効
	FALSE	DET 無効
デフォルト値	－	
コンフィギュレーションクラス		
プレコンパイルタイム		

本コンフィグパラメータの値を WDGM_DEV_ERROR_DETECT として参照可能とすること
【NWDGM118】.

3.10.2.4 WdgMImmediateReset

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMImmediateReset 【WDGM_Conf339】		
概要	アライブ監視異常時の即時リセット有効有無		
多重度	1		
型	EcucBooleanParamDef		
値の範囲	TRUE	即時リセット有効	
	FALSE	即時リセット無効	
デフォルト値	－		
コンフィギュレーションクラス			
プレコンパイルタイム			

本コンフィグパラメータの値を WDGM_IMMEDIATE_RESET として参照可能とすること
【NWDGM119】.

3.10.2.5 WdgMOffModeEnabled

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMOffModeEnabled 【WDGM_Conf340】	
概要	Watchdog Driver の OffMode 切替有効可否	
多重度	1	
型	EcucBooleanParamDef	
値の範囲	TRUE	OffMode 切替有効
	FALSE	OffMode 切替無効
デフォルト値	－	
コンフィギュレーションクラス		
プレコンパイルタイム		

本コンフィグパラメータの値を WDGM_OFF_MODE_ENABLED として参照可能とすること
【NWDGM120】.

3.10.2.6 WdgMDeadlineTimeoutEnabled

パラメータ名	/AUTOSAR/EcucDefs/WdgM/WdgMGeneral/WdgMDeadlineTimeoutEnabled【NWDGM_Conf077】	
概要	デッドライン監視のタイムアウト検出機能有効可否	
多重度	0..1	
型	EcucBooleanParamDef	
値の範囲	TRUE	デッドライン監視のタイムアウト検出機能有効
	FALSE	デッドライン監視のタイムアウト検出機能無効
デフォルト値	FALSE	
コンフィギュレーションクラス		
プレコンパイルタイム		

本コンフィグパラメータの値を WDG_M_DEADLINE_TIMEOUT_ENABLED として参照可能とすること【NWDGM121】.

3.10.2.7 WdgMMFCycleAndOsCounterSupervisionEnabled

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMMFCycleAndOsCounterSupervisionEnabled【NWDGM_Conf082】	
概要	WdgM_MainFunction コール周期/OS カウンタ監視機能有効可否	
多重度	0..1	
型	EcucBooleanParamDef	
値の範囲	TRUE	WdgM_MainFunction コール周期/OS カウンタ監視機能有効
	FALSE	WdgM_MainFunction コール周期/OS カウンタ監視機能無効
デフォルト値	ー	
コンフィギュレーションクラス		
プレコンパイルタイム		

本コンフィグパラメータの値を WDG_MF_CYCLE_AND_OSCOUNTER_SUPERVISION_ENABLED として参照可能とすること【NWDGM122】.

3.10.2.8 WdgMVersionInfoApi

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMVersionInfoApi 【WDGM_Conf302】	
概要	API WdgM_GetVersionInfo 有効可否. 無効の場合は不要なコードは除外すること.	
多重度	1	
型	EcucBooleanParamDef	
値の範囲	TRUE	API WdgM_GetVersionInfo 有効
	FALSE	API WdgM_GetVersionInfo 無効
デフォルト値	－	
コンフィギュレーションクラス		
プレコンパイルタイム		

本コンフィグパラメータの値を WDGM_VERSION_INFO_API として参照可能とすること
【NWDGM123】。

3.10.3 WdgMSupervisedEntity

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity 【WDGM_Conf303】	
概要	監視エンティティの全ての共通パラメータを含むコンテナ	
多重度	0..65535	
パラメータ	WdgMSupervisedEntityId	
	WdgMEcucPartitionRef	
	WdgMInternalCheckpointInitialRef	
	WdgMInternallCheckpointFinalRef	
	WdgMOsApplicationRef	
	WdgMOSCounter [NWDGM_Conf051]	
サブコンテナ	WdgMCheckpoint	
	WdgMInternalTransition	

3.10.3.1 WdgMSupervisedEntityId

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/ WdgMSupervisedEntityId 【WDGM_Conf304】
概要	監視エンティティの固有 ID
多重度	1
型	EcucIntegerParamDef
値の範囲	0..65535
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム	

本コンフィグパラメータのコンフィギュレーション内の最大値を WDGM_MAX_SEID として参照可能とすること 【NWDGM124】.

3.10.3.2 WdgMEcuPartitionRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/ WdgMEcuPartitionRef 【WDGM_Conf360】
概要	監視エンティティが実行されるパーティション参照 パーティションが停止した時, ECU リセットを避けるために監視エンティティも停止状態にしなければならない.
多重度	0..1
型	Reference to [EcucPartition]
コンフィギュレーションクラス	
プレコンパイルタイム	

3.10.3.3 WdgMInternalCheckpointInitialRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/ WdgMInternalCheckpointInitialRef 【WDGM_Conf343】
概要	監視エンティティの初期チェックポイント参照
多重度	0..1 【NWDGM008】
型	Reference to [WDGMCheckpoint]
コンフィギュレーションクラス	
プレコンパイルタイム	

AUTOSAR 仕様との違い

AUTOSAR 仕様ではパラメータ WdgMInternalCheckpointInitialRef の多重度は 1 となっているが、内部ロジカル監視を実施しないことも想定されるため、設定しないことを許可し、多重度を 0..1 とする。

3.10.3.4 WdgMInternalCheckpointFinalRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/ WdgMInternallCheckpointFinalRef 【WDGM_Conf344】
概要	監視エンティティの最終チェックポイント参照
多重度	0..65535 【NWDGM009】
型	Reference to [WDGMCheckpoint]
コンフィギュレーションクラス	
プレコンパイルタイム	

AUTOSAR 仕様との違い

AUTOSAR 仕様ではパラメータ WdgMInternallCheckpointFinalRef の多重度は 1..65535 となっているが、内部ロジカル監視を実施しないことも想定されるため、設定しないことを許可し、多重度を 0..65535 とする。

WdgMInternalCheckpointInitialRef と WdgMInternallCheckpointFinalRef が同じ参照先が設定されている場合、エラーとする 【NWDGM012】。

3.10.3.5 WdgMOsApplicationRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/ WdgMOsApplicationRef 【WDGM_Conf346】
概要	OS アプリケーション参照. 監視エンティティが WDGM_LOCAL_STATUS_FAILED になった時, OS アプリケーションのパーティションリスタートがトリガされる.
多重度	0..1
型	Reference to [OsApplication]
コンフィギュレーションクラス	
プレコンパイルタイム	

OsScalabilityClass が SC1 または SC2 の場合は STD_OFF, SC3 または SC4 の場合は STD_ON と
 して, OS アプリケーションのパーティションリスタートの有効可否を WDGM_OSAP_SUPPORT とし
 て参照可能とすること 【NWDGM126】.

Watchdog Manager コンフィギュレーションジェネレータは, OsScalabilityClass が SC3 または SC4
 以外に設定されており, WdgMOsApplicationRef が設定されているコンフィギュレーションをエラー
 とすること 【NWDGM131】.

3.10.3.6 WdgMOSCounter

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/ WdgMOSCounter 【NWDGM_Conf051】
概要	監視エンティティのデッドライン監視に用いる OS カウンタ
多重度	0..1
型	Reference to [OsCounter]
コンフィギュレーションクラス	
プレコンパイルタイム	

監視エンティティがデッドライン監視を行う場合, 本パラメータが設定されていない場合はエラーと
 する 【NWDGM044】.

3.10.4 WdgMCheckpoint

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/WdgMCheckpoint 【WDGM_Conf305】
概要	この監視エンティティの全てのチェックポイントを集められたコンテナ
多重度	1..65535
パラメータ	WdgMCheckpointId
サブコンテナ	—

3.10.4.1 WdgMCheckpointId

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/WdgMCheckpoint/WdgMCheckpointId 【WDGM_Conf306】
概要	チェックポイント固有 ID
多重度	1
型	EcucIntegerParamDef
値の範囲	0..65535
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム	

本パラメータは 0 から始まる連番で、同一の WdgMSupervisedEntity 内では重複のないよう設定すること【NWDGM061】。

3.10.5 WdgMInternalTransition

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/InternalTransition 【WDGM_Conf345】
概要	この監視エンティティの内部遷移のグラフを定義するコンテナ
多重度	0..65535
パラメータ	WdgMInternalTransitionDestRef
	WdgMInternalTransitionSourceRef
サブコンテナ	

コンテナ WdgMSupervisedEntity 内で以下のパラメータの片方でも設定されていない場合にコンテナ WdgMInternalTransition が設定されている場合はエラーとする 【NWDGM013】.

- WdgMInternalCheckpointInitialRef
- WdgMInternalCheckpointFinalRef

3.10.5.1 WdgMInternalTransitionDestRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/InternalTransition/ WdgMInternalTransitionDestRef 【WDGM_Conf351】
概要	この監視エンティティ内の内部遷移の遷移先チェックポイント
多重度	1
型	Reference to [WDGMCheckpoint]
コンフィギュレーションクラス	
プレコンパイルタイム	

コンテナ WdgMSupervisedEntity 内のパラメータ WdgMInternalCheckpointFinalRef に指定された参照先が、同コンテナ内のサブコンテナ WdgMInternalTransition 内のパラメータ WdgMInternalTransitionDestRef と同じ参照先が指定されているサブコンテナが一つも無い場合はエラーとする 【NWDGM015】.

3.10.5.2 WdgMInternalTransitionSourceRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMSupervisedEntity/InternalTransition/ WdgMInternalTransitionSourceRef 【WDGM_Conf350】
概要	この監視エンティティ内の内部遷移の遷移元チェックポイント
多重度	1
型	Reference to [WDGMCheckpoint]
コンフィギュレーションクラス	
プレコンパイルタイム	

コンテナ WdgMSupervisedEntity 内のパラメータ WdgMInternalCheckpointInitialRef に指定された参照先が、同コンテナ内のサブコンテナ WdgMInternalTransition 内のパラメータ WdgMInternalTransitionSourceRef と同じ参照先が指定されているサブコンテナが一つも無い場合はエラーとする 【NWDGM014】.

コンテナ WdgMSupervisedEntity 内のパラメータ WdgMInternalCheckpointFinalRef に指定された参照先が、同コンテナ内のサブコンテナ WdgMInternalTransition 内のパラメータ WdgMInternalTransitionSourceRef と同じ参照先が指定されているサブコンテナが一つでもある場合はエラーとする 【NWDGM017】.

3.10.6 WdgMWatchdog

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMWatchdog 【WDGM_Conf347】
概要	Watchdog Manager によってトリガされるための Watchdog の共通（モード非依存）パラメータを含むコンテナ.
多重度	0..65535
パラメータ	WdgMWatchdogName
	WdgMWatchdogDeviceRef
サブコンテナ	—

3.10.6.1 WdgMWatchdogName

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMWatchdog/WdgMWatchdogName 【WDGM_Conf348】
概要	Watchdog インスタンスを象徴する名前
多重度	1
型	EcucStringParamDef
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム	

3.10.6.2 WdgMWatchdogDeviceRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMWatchdog/WdgMWatchdogDeviceRef 【WDGM_Conf349】
概要	Watchdog Interface のあるデバイスコンテナの参照. 参照されたコンテナ WdgIfDevice の中のパラメータ WdgIfDeviceIndex は Watchdog Manager が Watchdog インスタンスへ要求する WdgIf_SetTriggerCondition のために使用するインデックスパラメータを含む.
多重度	1
型	Reference to [WDGIfDevice]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム	

WDGM_[本コンフィグパラメータ名]として、WdgMWatchdogDeviceRef の参照先の WdgIfDeviceIndex を参照可能とすること 【NWDGM125】.

3.10.7 WdgMConfigSet

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet 【WDGM_Conf337】
概要	Watchdog Manager の複数コンフィギュレーションの内の一つを記述するコンテナ。 これは多重コンフィギュレーションコンテナである。すなわち、このコンテナとサブコンテナはコンフィギュレーションセットの 1 つずつに存在する。
多重度	1..* 【NWDGM007】
パラメータ	WdgMInitialMode
サブコンテナ	WdgMDemEventParameterRefs
	WdgMMode

本コンフィグパラメータのコンフィギュレーション内の多重度を WDGM_CONFIG_NUM として参照可能とすること 【NWDGM128】。

AUTOSAR 仕様との違い

AUTOSAR 仕様ではコンテナ WdgMConfigSet の多重度は 1 となっているが、ポストビルド対応するため、多重度を 1..* とする。

3.10.7.1 WdgMInitialMode

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMInitialMode 【WDGM_Conf336】
概要	Watchdog Manager 初期化後のモード
多重度	1
型	Reference to [WDGMMode]
	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム、ポストビルドタイム	

3.10.8 WdgMDemEventParametersRefs

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMDemEventParameterRefs 【WDGM_Conf353】
概要	付随するエラーが起こった場合に、API Dem_ReportErrorStatus を使用して起動される DemEventParameter の要素を参照する要素のためのコンテナ。EventId は参照された DemEventParameter の DemEventId の値から取得する。標準化されたエラーはコンテナによって提供され、ベンダー固有のエラー参照により拡張することができる。
多重度	0..1
パラメータ	WDGM_E_IMPROPER_CALLER
	WDGM_E_SUPERVISION [NWDGM_Conf045]
	WDGM_E_SET_MODE
	WDGM_E_OSCOUNTER [NWDGM_Conf106]
	WDGM_E_MFCYCLE_OR_OSCOUNTER [NWDGM_Conf083]
サブコンテナ	—

本コンテナに所属するパラメータのうち設定されていないパラメータがある場合、該当するパラメータに対応する条件で API Dem_ReportErrorStatus をコールしない【NWDGM004】。

3.10.8.1 WDGM_E_IMPROPER_CALLER

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMDemEventParameterRefs/ WDGM_E_IMPROPER_CALLER 【WDGM_Conf357】
概要	defensive behavior が不適切なコールを検出したとき、発行するものとする DevEventParameter の参照
多重度	0..1
型	Reference to [DemEventParameter]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム、ポストビルドタイム	

3.10.8.2 WDMG_E_SUPERVISION

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMDemEventParameterRefs/ WDMG_E_SUPERVISION 【NWDGM_Conf045】
概要	モニタでフェイルになり Watchdog がリセットすることが発生したとき、発行するものとする DemEventParameter の参照
多重度	0..1
型	Reference to [DemEventParameter]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム、ポストビルドタイム	

3.10.8.3 WDMG_E_SET_MODE

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMDemEventParameterRefs/ WDMG_E_SET_MODE 【WDMG_Conf355】
概要	Watchdog Driver のモード切替がフェイルとなったエラーが発生したとき、発行するものとする DemEventParameter の参照
多重度	0..1
型	Reference to [DemEventParameter]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム、ポストビルドタイム	

3.10.8.4 Wdgm_E_OSCOUNTER

パラメータ名	/AUTOSAR/EcucDefs/WdgM/WdgMConfigSet/WdgMDemEventParameterRefs/Wdgm_E_OSCOUNTER 【NWDGM_Conf106】
概要	OS カウンタのサービス(GetCounterValue, GetElapsedValue)の返り値が E_OK でなかった場合、発行するものとする DemEventParameter の参照
多重度	0..1
型	Reference to [DemEventParameter]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム、ポストビルドタイム	

OS カウンタのサービス(GetCounterValue, GetElapsedValue)の返り値が E_OK でなかった場合、Wdgm_E_OSCOUNTER として DEM を発行し、グローバル監視状態を Wdgm_GLOBAL_STATUS_STOPPED に遷移するものとする 【NWDGM093】.

3.10.8.5 Wdgm_E_MFCYCLE_OR_OSCOUNTER

パラメータ名	/AUTOSAR/EcucDefs/WdgM/WdgMConfigSet/WdgMDemEventParameterRefs/Wdgm_E_MFCYCLE_OR_OSCOUNTER 【NWDGM_Conf083】
概要	WdgM_MainFunction コール周期/OS カウンタ監視機能が異常を検出したとき、発行するものとする DemEventParameter の参照
多重度	0..1
型	Reference to [DemEventParameter]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム、ポストビルドタイム	

Wdgm_E_MFCYCLE_OR_OSCOUNTER は WdgMMFCycleAndOsCounterSupervisionEnabled が FALSE の状態で設定された場合、ジェネレータは警告を発行する 【NWDGM090】.

3.10.9 WdgMMode

コンテナ名	/AUTOSAR/EcucDefs/WdgM/WdgMConfigSet/WdgMMode 【WDGM_Conf335】
概要	Watchdog Manager の複数あるモードの内の 1 つを示したコンテナ
多重度	1..255
パラメータ	WdgMExpiredSupervisionCycleTol
	WdgMModeId
	WdgMSupervisionCycle
サブコンテナ	WdgMAliveSupervision
	WdgMDeadlineSupervision
	WdgMExternalLogicalSupervision
	WdgMLocalStatusParams
	WdgMTrigger
	WdgMMFCycleAndOsCounterSupervision [NWDGM_Conf077]

3.10.9.1 WdgMExpiredSupervisionCycleTol

パラメータ名	/AUTOSAR/EcucDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMExpiredSupervisionCycleTol 【WDGM_Conf329】
概要	グローバル監視状態が EXPIRED に到達した後、Watchdog トリガのブロックをどれだけ長く延期させるかのための EXPIRED となった監視サイクルの量を決定するための値を定義するために使用されるパラメータ。
多重度	1
型	EcucIntegerParamDef
値の範囲	0..65535
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

3.10.9.2 WdgMModeId

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMModeId 【WDGM_Conf307】
概要	モードの ID を決定するパラメータ。この ID は例えばサービス WdgM_SetMode にパラメータとして渡される。
多重度	1
型	EcucIntegerParamDef
値の範囲	0..255
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

本パラメータは 0 から始まる連番で、同一の WdgMConfigSet の中では重複のないよう設定すること【NWDGM062】。

3.10.9.3 WdgMSupervisionCycle

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMSupervisionCycle 【WDGM_Conf330】
概要	WdgM_MainFunction のスケジュール周期を定義するパラメータ。単位[s]。
多重度	1
型	EcucFloatParamDef
値の範囲	0..INF
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

本コンフィグパラメータは周期関数のコール周期であるため、コンフィグパラメータとして 0 を与えられた場合、エラーとする【NWDGM129】。

3.10.10 WdgMAliveSupervision

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMAliveSupervision 【WDGM_Conf308】
概要	あるチェックポイントのアライブ監視の全てのコンフィギュレーションパラメータを含むコンテナ。それぞれのチェックポイントは異なったパラメータを持つかもしれない。例えば、異なった最小最大マージンを持つかもしれない。
多重度	0..65535
パラメータ	WdgMExpectedAliveIndications
	WdgMMaxMargin
	WdgMMinMargin
	WdgMSupervisionReferenceCycle
	WdgMAliveSupervisionCheckpointRef
サブコンテナ	—

3.10.10.1 WdgMExpectedAliveIndications

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMAliveSupervision/WdgMExpectedAliveIndications 【WDGM_Conf311】
概要	参照された対応監視エンティティに従って定義された監視サイクル量の範囲内で期待されるチェックポイントのアライブ通知量のパラメータ。
多重度	1
型	EcucIntegerParamDef
値の範囲	0..65535
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム，ポストビルドタイム	

3.10.10.2 WdgMMaxMargin

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMAliveSupervision/ WdgMMaxMargin 【WDGM_Conf313】
概要	対応する監視参照サイクル内で期待されるアライブ通知に加えて許可されるチェックポイントのアライブ通知の量のパラメータ。
多重度	1
型	EcucIntegerParamDef
値の範囲	0..255
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

WdgMExpectedAliveIndications + WdgMMaxMargin の結果が 65535 を上回る場合, エラーとする【NWDGM108】.

3.10.10.3 WdgMMinMargin

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMAliveSupervision/ WdgMMinMargin 【WDGM_Conf312】
概要	対応する監視参照サイクル内で期待されるアライブ通知から欠けることが許可されるチェックポイントのアライブ通知の量のパラメータ。
多重度	1
型	EcucIntegerParamDef
値の範囲	0..255
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

WdgMMinMargin が WdgMExpectedAliveIndications を上回る場合, エラーとする【NWDGM109】.

3.10.10.4 WdgMSupervisionReferenceCycle

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMAliveSupervision/WdgMSupervisionReferenceCycle 【WDGM_Conf310】
概要	対応する監視エンティティに従ってカウントされたアライブ通知を検査するためのアライブ監視メカニズムによって参照として使用される監視サイクル量のパラメータ。
多重度	1
型	EcucIntegerParamDef
値の範囲	1..65535
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム，ポストビルドタイム	

3.10.10.5 WdgMAliveSupervisionCheckpointRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMAliveSupervision/WdgMAliveSupervisionCheckpointRef 【WDGM_Conf309】
概要	監視されなければならない監視エンティティ内のチェックポイントの参照
多重度	1
型	Reference to [WDGMCheckpoint]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム，ポストビルドタイム	

WdgMAliveSupervisionCheckpointRef が所属するモードによって参照される監視エンティティ (WdgMLocalStatusSupervisedEntityRef) のうち、いずれにも該当しない監視エンティティに所属するチェックポイントを参照した場合、エラーとする 【NWDGM063】。

3.10.11 WdgMDeadlineSupervision

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMDeadlineSupervision 【WDGM_Conf314】
概要	ある監視エンティティのためのデッドライン監視の全てのコンフィギュレーションパラメータを含むコンテナ
多重度	0..65535
パラメータ	WdgMDeadlineMax
	WdgMDeadlineMin
	WdgMDeadlineStartRef
	WdgMDeadlineStopRef
	WdgMDeadlineTimeoutDetect [NWDGM_Conf075]
サブコンテナ	—

WdgMDeadlineStartRef と WdgMDeadlineStopRef が同じチェックポイントを参照した場合、エラーとする 【NWDGM019】.

WdgMDeadlineMax < WdgMDeadlineMin となるような設定を与えた場合、エラーとする 【NWDGM107】.

3.10.11.1 WdgMDeadlineMax

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMDeadlineSupervision/WdgMDeadlineMax 【WDGM_Conf318】
概要	デッドラインが満たされていると見なされる最長時間間隔のパラメータ. 単位[s].
多重度	1
型	EcucFloatParamDef
値の範囲	0..INF
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

3.10.11.2 WdgMDeadlineMin

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMDeadlineSupervision/ WdgMDeadlineMin 【WDGM_Conf317】
概要	デッドラインが満たされていると見なされる最短時間間隔のパラメータ。単位[s]。
多重度	1
型	EcucFloatParamDef
値の範囲	0..INF
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム，ポストビルドタイム	

3.10.11.3 WdgMDeadlineStartRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMDeadlineSupervision/WdgMDeadlineStartRef 【WDGM_Conf315】
概要	デッドライン監視の開始チェックポイント参照
多重度	1
型	Reference to [WdgMCheckpoint]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム，ポストビルドタイム	

WdgMDeadlineStartRef が所属するモードによって参照される監視エンティティ (WdgMLocalStatusSupervisedEntityRef) のうち、いずれにも該当しない監視エンティティに所属するチェックポイントを参照した場合、エラーとする 【NWDGM064】。

3.10.11.4 WdgMDeadlineStopRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMDeadlineSupervision/ WdgMDeadlineStopRef 【WDGM_Conf316】
概要	デッドライン監視の停止チェックポイント参照
多重度	1
型	Reference to [WDGMCheckpoint]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

WdgMDeadlineStopRef が所属するモードによって参照される監視エンティティ (WdgMLocalStatusSupervisedEntityRef)のうち、いずれにも該当しない監視エンティティに所属するチェックポイントを参照した場合、エラーとする 【NWDGM065】.

3.10.11.5 WdgMDeadlineTimeoutDetect

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMDeadlineSupervision/WdgMDeadlineTimeoutDetect 【NWDGM_Conf075】	
概要	デッドライン監視のタイムアウト検出有効可否	
多重度	0..1	
型	EcucBooleanParamDef	
値の範囲	TRUE	デッドライン監視のタイムアウト検出有効
	FALSE	デッドライン監視のタイムアウト検出無効
デフォルト値	FALSE	
コンフィギュレーションクラス		
プレコンパイルタイム，ポストビルドタイム		

WdgMDeadlineTimeoutDetect は WdgMDeadlineTimeoutEnabled が FALSE の状態で設定された場合、ジェネレータは警告を発行する 【NWDGM092】

3.10.12 WdgMExternalLogicalSupervision

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMExternalLogicalSupervision 【WDGM_Conf319】
概要	一つの外部グラフのロジカル監視の全てのコンフィギュレーションパラメータを含むコンテナ
多重度	0..65535
パラメータ	WdgMExternalCheckpointFinalRef
	WdgMExternalCheckpointInitialRef
サブコンテナ	WdgMExternalTransition

コンテナ WdgMExternalLogicalSupervision 内の以下のパラメータに同じ値が指定されたコンテナが存在する場合はエラーとする 【NWDGM020】.

- WdgMExternalCheckpointFinalRef
- WdgMExternalCheckpointInitialRef

3.10.12.1 WdgMExternalCheckpointFinalRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMExternalLogicalSupervision/ WdgMExternalCheckpointFinalRef 【WDGM_Conf324】
概要	この外部グラフの最終チェックポイント参照
多重度	1..65535
型	Reference to [WDGMCheckpoint]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

WdgMExternalCheckpointFinalRef が所属するモードによって参照される監視エンティティ (WdgMLocalStatusSupervisedEntityRef)のうち、いずれにも該当しない監視エンティティに所属するチェックポイントを参照した場合、エラーとする 【NWDGM066】.

3.10.12.2 WdgMExternalCheckpointInitialRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMExternalLogicalSupervision/WdgMExternalCheckpointInitialRef 【WDGM_Conf323】
概要	この外部グラフの初期チェックポイント参照
多重度	1..65535
型	Reference to [WDGMCheckpoint]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

WdgMExternalCheckpointInitialRef が所属するモードによって参照される監視エンティティ (WdgMLocalStatusSupervisedEntityRef) のうち、いずれにも該当しない監視エンティティに所属するチェックポイントを参照した場合、エラーとする 【NWDGM067】.

3.10.12.3 WdgMExternalTransition

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMExternalLogicalSupervision/WdgMExternalTransition 【WDGM_Conf320】
概要	監視エンティティを超えた外部遷移のためのチェックポイントをまとめたコンテナ
多重度	1..65535 【NWDGM133】
パラメータ	WdgMExternalTransitionDestRef
	WdgMExternalTransitionSourceRef
サブコンテナ	—

3.10.12.3.1 WdgMExternalTransitionDestRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMExternalLogicalSupervision/WdgMExternalTransition/ WdgMExternalTransitionDestRef 【WDGM_Conf322】
概要	外部遷移の遷移先のチェックポイント参照
多重度	1
型	Reference to [WDGMCheckpoint]
値の範囲	－
デフォルト値	－
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

コンテナ WdgMExternalLogicalSupervision 内のパラメータ WdgMExternalCheckpointFinalRef に指定された参照先が、同コンテナ内のサブコンテナ WdgMExternalTransition 内のパラメータ WdgMExternalTransitionDestRef と同じ参照先が指定されているサブコンテナが一つも無い場合はエラーとする【NWDGM022】.

WdgMExternalTransitionDestRef が所属するモードによって参照される監視エンティティ (WdgMLocalStatusSupervisedEntityRef)のうち、いずれにも該当しない監視エンティティに所属するチェックポイントを参照した場合、エラーとする【NWDGM068】.

3.10.12.3.2 WdgMExternalTransitionSourceRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMExternalLogicalSupervision/WdgMExternalTransition/ WdgMExternalTransitionSourceRef 【WDGM_Conf321】
概要	外部遷移の遷移元のチェックポイント参照
多重度	1
型	Reference to [WDGMCheckpoint]
値の範囲	－
デフォルト値	－
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

コンテナ WdgMExternalLogicalSupervision 内のパラメータ WdgMExternalCheckpointInitialRef に指定された参照先が、同コンテナ内のサブコンテナ WdgMExternalTransition 内のパラメータ WdgMExternalTransitionSourceRef と同じ参照先が指定されているサブコンテナが一つも無い場合はエラーとする【NWDGM021】.

コンテナ WdgMExternalLogicalSupervision 内のパラメータ WdgMExternalCheckpointFinalRef に指定された参照先が、同コンテナ内のサブコンテナ WdgMExternalTransition 内のパラメータ WdgMExternalTransitionSourceRef と同じ参照先が指定されているサブコンテナが一つでもある場合はエラーとする【NWDGM024】.

WdgMExternalTransitionSourceRef が所属するモードによって参照される監視エンティティ (WdgMLocalStatusSupervisedEntityRef)のうち、いずれにも該当しない監視エンティティに所属するチェックポイントを参照した場合、エラーとする【NWDGM069】.

3.10.13 WdgMTrigger

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMTrigger 【WDGM_Conf331】
概要	Watchdog タイマのトリガの全てのコンフィギュレーションパラメータをまとめたコンテナ
多重度	0..255
パラメータ	WdgMTriggerConditionValue
	WdgMWatchdogMode
	WdgMTriggerWatchdogRef
サブコンテナ	—

3.10.13.1 WdgMTriggerConditionValue

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMTrigger/WdgMTriggerConditionValue 【WDGM_Conf333】
概要	この Watchdog のために WdgIf_SetTriggerCondition に渡す値を含むパラメータ
多重度	1
型	EcucIntegerParamDef
値の範囲	1..65535
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

3.10.13.2 WdgMWatchdogMode

パラメータ名	/AUTOSAR/EcucDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMTrigger/WdgMWa tchdogMode 【WDGM_Conf332】	
概要	Watchdog Manager mode 内の参照 Watchdog のために使用される Watchdog mode を 含むパラメータ．実装型 WdgIf_ModeType．	
多重度	1	
型	EcucEnumerationParamDef	
値の範囲	WDGIF_FAST_MODE	—
	WDGIF_SLOW_MODE	—
	WDGIF_OFF_MODE	—
デフォルト値	—	
コンフィギュレーションクラス		
プレコンパイルタイム，ポストビルドタイム		

WdgMOffModeEnabled が FALSE であるとき，WdgMWatchdogMode に WDGIF_OFF_MODE を設定された場合，エラーとする 【NWDGM039】.

3.10.13.3 WdgMTriggerWatchdogRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMTrigger/WdgMTriggerWatchdogRef 【WDGM_Conf334】
概要	構成される Watchdog の参照
多重度	1
型	Reference to [WdgMWatchdog]
値の範囲	－
デフォルト値	－
コンフィギュレーションクラス	
プレコンパイルタイム，ポストビルドタイム	

3.10.14 WdgMMFCycleAndOsCounterSupervision

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/ WdgMMFCycleAndOsCounterSupervision 【NWDGM_Conf078】
概要	WdgM_MainFunction コール周期/OS カウンタ監視機能を設定するコンテナ
多重度	0..1
パラメータ	WdgMMFCycleMinTol
	WdgMMFCycleMaxTol
	WdgMOsCounterRef
サブコンテナ	—

WdgMMFCycleAndOsCounterSupervision は WdgMMFCycleAndOsCounterSupervisionEnabled が FALSE の状態で設定された場合、ジェネレータは警告を発行する 【NWDGM091】.

3.10.14.1 WdgMMFCycleMinTol

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/ WdgMMFCycleAndOsCounterSupervision/WdgMMFCycleMinTol 【NWDGM_Conf079】
概要	WdgM_MainFunction コール周期の許容下限. 単位[s].
多重度	1
型	EcucFloatParamDef
値の範囲	0..INF
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

3.10.14.2 WdgMMFCycleMaxTol

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/ WdgMMFCycleAndOsCounterSupervision/WdgMMFCycleMaxTol 【NWDGM_Conf080】
概要	WdgM_MainFunction コール周期の許容上限. 単位[s].
多重度	1
型	EcucFloatParamDef
値の範囲	0..INF
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

3.10.14.3 WdgMOsCounterRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/ WdgMMFCycleAndOsCounterSupervision/WdgMOsCounterRef 【NWDGM_Conf081】
概要	監視に使用する，および監視対象となる OS カウンタへの参照
多重度	1..65535
型	EcucFloatParamDef
値の範囲	Reference to [OsCounter]
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム，ポストビルドタイム	

3.10.15 WdgMLocalStatusParams

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMLocalStatusPara ms 【WDGM_Conf325】
概要	監視エンティティのローカル状態のための全てのコンフィギュレーションパラメータ を含むコンテナ
多重度	0..65535
パラメータ	WdgMFailedAliveSupervisionRefCycleTol
	WdgMLocalStatusSupervisedEntityRef
サブコンテナ	—

3.10.15.1 WdgMFailedAliveSupervisionRefCycleTol

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMLocalStatusPara ms/ WdgMFailedAliveSupervisionRefCycleTol 【WDGM_Conf327】
概要	この監視エンティティのための不正/失敗したアライブ監視の参照サイクルの許容量
多重度	1
型	EcucIntegerParamDef
値の範囲	0..255
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム，ポストビルドタイム	

3.10.15.2 WdgMLocalStatusSupervisedEntityRef

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMConfigSet/WdgMMode/WdgMLocalStatusParams/WdgMLocalStatusSupervisedEntityRef 【WDGM_Conf326】
概要	ローカル状態のパラメータを指定した監視エンティティの参照
多重度	1
型	Reference to [WDGMSupervisedEntity]
値の範囲	—
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム, ポストビルドタイム	

3.10.16 WdgMCallerIds

コンテナ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMCallerIds 【WDGM_Conf358】
概要	WdgM_SetMode をコールすることを許可されたコール元のための正しい CallerId の定義を含むコンテナ
多重度	0..1
パラメータ	WdgMCallerId
サブコンテナ	—

3.10.16.1 WdgMCallerId

パラメータ名	/AUTOSAR/EcuDefs/WdgM/WdgMGeneral/WdgMCallerIds/WdgMCallerId 【WDGM_Conf359】
概要	WdgM_SetMode をコールすることを許可されたコール元のための正しい一つの CallerId の定義
多重度	0..255
型	EcucIntegerParamDef
値の範囲	0..65535
デフォルト値	—
コンフィギュレーションクラス	
プレコンパイルタイム	

本コンフィグパラメータのコンフィギュレーション内の多重度を WDGM_CALLER_ID_NUM として参照可能とすること 【NWDGM127】.

4 リファレンス

4.1 データ型一覧

表 4-1 データ型一覧

データ型名	概要
WdgM_ConfigType	Watchdog Manager のコンフィギュレーション型
WdgM_SupervisedEntityIdType	監視エンティティ ID 型
WdgM_CheckpointIdType	チェックポイント型
WdgM_ModeType	Watchdog Manager モード型
WdgM_LocalStatusType	ローカル監視状態型
WdgM_GlobalStatusType	グローバル監視状態型

4.2 定数一覧

表 4-2 定数一覧

定数名	概要
WDGM_LOCAL_STATUS_OK	監視エンティティが正常であることを示す定数
WDGM_LOCAL_STATUS_FAILED	監視エンティティが復帰可能な異常であることを示す定数
WDGM_LOCAL_STATUS_EXPIRED	監視エンティティが復帰不可能な異常であることを示す定数
WDGM_LOCAL_STATUS_DEACTIVATED	監視エンティティが一時的に停止中であることを示す定数
WDGM_GLOBAL_STATUS_OK	監視が正常であることを示す定数
WDGM_GLOBAL_STATUS_FAILED	監視が復帰可能な異常であることを示す定数
WDGM_GLOBAL_STATUS_EXPIRED	監視が復帰不可能な異常であるがトリガ操作実施中であることを示す定数
WDGM_GLOBAL_STATUS_STOPPED	監視が復帰不可能な異常かつトリガ操作停止中であることを示す定数
WDGM_GLOBAL_STATUS_DEACTIVATED	Watchdog Manager が初期化未完了であり Watchdog の管理がされていないことを示す定数

4.3 API 一覧

表 4-3 API 一覧

システムサービス名	概要
WdgM_Init	初期化
WdgM_DeInit	非初期化
WdgM_GetVersionInfo	バージョン情報取得
WdgM_SetMode	Watchdog Manager モード設定
WdgM_GetMode	Watchdog Manager モード取得
WdgM_CheckpointReached	チェックポイント到達通知
WdgM_UpdateAliveCounter	アライブカウンタ更新通知
WdgM_GetLocalStatus	監視エンティティのローカル監視状態取得
WdgM_GetGlobalStatus	監視エンティティのグローバル監視状態取得
WdgM_PerformReset	Watchdog リセット指示要求
WdgM_GetFirstExpiredSEID	最初に WDGM_LOCAL_STATUS_EXPIRED の状態となった監視エンティティの ID 取得

4.4 コールバック関数一覧

なし

4.5 周期関数一覧

表 4-4 周期関数一覧

周期関数名	概要
WdgM_MainFunction	Watchdog Manager の周期関数

4.6 エラーコード一覧

表 4-5 エラーコード一覧

エラーコード名	概要
WDGM_E_SUPERVISION	監視が異常ありかつ Watchdog のリセット確定
WDGM_E_SET_MODE	Watchdog Driver のモード切替失敗
WDGM_E_IMPROPER_CALLER	不正なコール検出
WDGM_E_NO_INIT	初期化が未実施
WDGM_E_PARAM_CONFIG	コンフィギュレーション設定が不正
WDGM_E_PARAM_MODE	パラメータ mode が不正
WDGM_E_PARAM_SEID	パラメータ SEID が不正
WDGM_E_INV_POINTER	NULL ポインタが指定された
WDGM_E_DISABLE_NOT_ALLOWED	Watchdog の無効化が許可されていない
WDGM_E_CPID	不正なチェックポイント ID
WDGM_E_DEPRECATED	非推奨の API が使用された
WDGM_E_AMBIGIOUS	対象のアライブ監視が特定できない
WDGM_E_SEDEACTIVATED	非アクティブな監視エンティティのチェックポイントが指定された

4.7 仕様番号一覧

本章は特記すべき仕様タグについて述べるものである。本章に記載されていない仕様タグであっても、本書中に記載した仕様タグはすべて漏れなく実装される。

4.7.1 削除した AUTOSAR 仕様一覧

仕様タグ	削除理由
<WDGM006>	仕様矛盾解消のため
<WDGM013>	バージョンチェックはユーザが行うため(※R4.1.1 で削除される)
<WDGM015>	仕様矛盾解消のため
<WDGM027>	非推奨なシステムサービス(WdgM_UpdateAliveCounter)を削除するため
<WDGM028>	非推奨なシステムサービス(WdgM_UpdateAliveCounter)を削除するため
<WDGM030>	〔NWDGM039〕により不要となるため
<WDGM031>	〔NWDGM039〕により不要となるため
<WDGM155>	非推奨なシステムサービス(WdgM_UpdateAliveCounter)を削除するため
<WDGM273>	〔WDGM332〕に含まれるため
<WDGM290>	非推奨なシステムサービス(WdgM_UpdateAliveCounter)を削除するため
<WDGM308>	内部グラフは1つしか構成できないため
<WDGM317>	仕様矛盾解消のため
<WDGM318>	非推奨なシステムサービス(WdgM_UpdateAliveCounter)を削除するため
<WDGM320>	非推奨なシステムサービス(WdgM_UpdateAliveCounter)を削除するため
<WDGM329>	〔WDGM331〕に含まれるため
<WDGM345>	仕様ではないため

4.7.2 改変した AUTOSAR 仕様一覧

仕様タグ	改変内容
〔NWDGM001〕	トリガコンディション設定ルール変更 〔WDGM223〕
〔NWDGM007〕	WdgMConfigSet 多重度変更 〔WDGM_Conf337〕
〔NWDGM008〕	WdgMInternalCheckpointInitialRef 多重度変更 〔WDGM_Conf343〕
〔NWDGM009〕	WdgMInternalCheckpointFinalRef 多重度変更 〔WDGM_Conf344〕
〔NWDGM042〕	Dem_ReportError コール条件変更 〔WDGM142_1〕
〔NWDGM_Conf045〕	WDGM_E_SUPERVISION と WDGM_E_MONITORING の統一 〔WDGM_Conf354〕
〔NWDGM046〕	CallerID 確認条件のコンフィグパラメータを WdgMDevErrorDetect から WdgMDefensiveBehavior に変更 〔WDGM245〕
〔NWDGM047〕	初期化確認条件のコンフィグパラメータを WdgMDefensiveBehavior から WdgMDevErrorDetect に変更 〔WDGM275〕

〔NWDGM048〕	グローバル監視状態遷移時アクション追加〔WDGM077〕
〔NWDGM049〕	グローバル監視状態遷移条件変更〔WDGM219〕
〔NWDGM050〕	グローバル監視状態遷移条件変更〔WDGM220〕
〔NWDGM053〕	デッドライン監視開始判定条件変更〔WDGM229〕
〔NWDGM054〕	デッドライン監視開始判定条件変更〔WDGM229〕
〔NWDGM055〕	デッドライン監視開始判定条件変更〔WDGM229〕
〔NWDGM105〕	グローバル監視状態遷移時アクション追加〔WDGM215〕
〔NWDGM133〕	WdgMExternalTransition 多重度変更〔WDGM_Conf320〕

4.7.3 追加した NCES 仕様一覧

仕様タグ	概要
〔NWDGM004〕	Dem_ReportError コール条件
〔NWDGM012〕	コンフィグパラメータ設定制約
〔NWDGM013〕	コンフィグパラメータ設定制約
〔NWDGM014〕	コンフィグパラメータ設定制約
〔NWDGM015〕	コンフィグパラメータ設定制約
〔NWDGM017〕	コンフィグパラメータ設定制約
〔NWDGM019〕	コンフィグパラメータ設定制約
〔NWDGM020〕	コンフィグパラメータ設定制約
〔NWDGM021〕	コンフィグパラメータ設定制約
〔NWDGM022〕	コンフィグパラメータ設定制約
〔NWDGM024〕	コンフィグパラメータ設定制約
〔NWDGM039〕	コンフィグパラメータ設定制約
〔NWDGM043〕	Dem_ReportErrorStatus コール時の引数 EventStatus の値
〔NWDGM044〕	コンフィグパラメータ設定制約
〔NWDGM_Conf051〕	WdgMOSCounter 追加
〔NWDGM052〕	ポストビルドタイムパラメータのセレクタブルのみ対応を定義
〔NWDGM059〕	WdgM_GetGlobalStatus コール条件
〔NWDGM060〕	コンフィグパラメータ設定制約
〔NWDGM061〕	コンフィグパラメータ設定制約
〔NWDGM062〕	コンフィグパラメータ設定制約
〔NWDGM063〕	コンフィグパラメータ設定制約
〔NWDGM064〕	コンフィグパラメータ設定制約
〔NWDGM065〕	コンフィグパラメータ設定制約
〔NWDGM066〕	コンフィグパラメータ設定制約
〔NWDGM067〕	コンフィグパラメータ設定制約

仕様タグ	概要
[NWDGM068]	コンフィグパラメータ設定制約
[NWDGM069]	コンフィグパラメータ設定制約
[NWDGM070]	WdgM_PerformReset コール後動作
[NWDGM071]	WdgM_PerformReset コール後動作
[NWDGM073]	WdgM_PerformReset コール後動作
[NWDGM074]	WdgM_PerformReset コール後動作
[NWDGM_Conf075]	WdgMDeadlineTimeoutDetect 追加
[NWDGM076]	デッドライン監視タイムアウト検出機能
[NWDGM_Conf077]	WdgMDeadlineTimeoutEnable 追加
[NWDGM_Conf078]	WdgMMFCycleAndOsCounterSupervision 追加
[NWDGM_Conf079]	WdgMMFCycleMinTol 追加
[NWDGM_Conf080]	WdgMMFCycleMaxTol 追加
[NWDGM_Conf081]	WdgMOsCounterRef 追加
[NWDGM_Conf082]	WdgMMFCycleAndOsCounterSupervisionEnabled 追加
[NWDGM_Conf083]	WDGM_E_MFCYCLE_OR_OSCOUNTER 追加
[NWDGM084]	グローバル監視状態遷移
[NWDGM085]	初期化处理
[NWDGM086]	初期化处理
[NWDGM087]	必須インタフェース追加
[NWDGM088]	タイムスタンプ取得先
[NWDGM089]	WdgM_SetMode 内 WdgIf_SetTriggerCondition コール処理
[NWDGM090]	コンフィグパラメータ設定警告
[NWDGM091]	コンフィグパラメータ設定警告
[NWDGM092]	コンフィグパラメータ設定警告
[NWDGM093]	WDGM_E_OSCOUNTER 報告条件
[NWDGM094]	WdgM_MainFunction コール周期/OS カウンタ監視機能
[NWDGM095]	WdgM_MainFunction コール周期/OS カウンタ監視機能
[NWDGM096]	WdgM_MainFunction コール周期/OS カウンタ監視機能
[NWDGM097]	WdgM_MainFunction コール周期/OS カウンタ監視機能
[NWDGM098]	WdgM_MainFunction コール周期/OS カウンタ監視機能
[NWDGM099]	WdgM_MainFunction コール周期/OS カウンタ監視機能
[NWDGM100]	WdgM_MainFunction コール周期/OS カウンタ監視機能
[NWDGM101]	WdgM_MainFunction コール周期/OS カウンタ監視機能
[NWDGM102]	API 同時コール制約
[NWDGM103]	API 同時コール制約

仕様タグ	概要
〔NWDGM104〕	API 同時コール制約
〔NWDGM_Conf106〕	WDGM_E_OSOUNTER 追加
〔NWDGM107〕	コンフィグパラメータ設定制約
〔NWDGM108〕	コンフィグパラメータ設定制約
〔NWDGM109〕	コンフィグパラメータ設定制約
〔NWDGM110〕	ファイル概要
〔NWDGM111〕	ファイル概要
〔NWDGM112〕	ファイル概要
〔NWDGM113〕	ファイル概要
〔NWDGM114〕	コンフィグパラメータ設定制約
〔NWDGM115〕	コンフィグパラメータ設定制約
〔NWDGM116〕	WdgM ジェネレータ生成定数
〔NWDGM117〕	WdgM ジェネレータ生成定数
〔NWDGM118〕	WdgM ジェネレータ生成定数
〔NWDGM119〕	WdgM ジェネレータ生成定数
〔NWDGM120〕	WdgM ジェネレータ生成定数
〔NWDGM121〕	WdgM ジェネレータ生成定数
〔NWDGM122〕	WdgM ジェネレータ生成定数
〔NWDGM123〕	WdgM ジェネレータ生成定数
〔NWDGM124〕	WdgM ジェネレータ生成定数
〔NWDGM125〕	WdgM ジェネレータ生成定数
〔NWDGM126〕	WdgM ジェネレータ生成定数
〔NWDGM127〕	WdgM ジェネレータ生成定数
〔NWDGM128〕	WdgM ジェネレータ生成定数
〔NWDGM129〕	コンフィグパラメータ設定制約
〔NWDGM130〕	コンフィグパラメータ設定制約
〔NWDGM131〕	コンフィグパラメータ設定制約
〔NWDGM132〕	ロジカル監視補足説明

変更履歴

バージョン	変更日	変更内容	変更者
1.0.0	2016/3/31	新規作成	NCES
1.0.1	2016/8/10	NWDGM055 誤記修正 WDGM021 誤記修正 WDGM255 誤記修正 WDGM256 誤記修正 WDGM284 誤記修正 WDGM347 返り値として E_NOT_OK を返すことを追記 WDGM212_3 削除 WDGM132 追加 WDGM020 誤記修正 NWDGM096 誤記修正 NWDGM076 誤記修正 拡張エラーコード一覧に WDGM_E_CONCURRENTLY_CALLED 追加	NCES
1.0.2	2017/1/31	WdgM_Conf078→NWdgM_Conf078 に修正 WDGM283 誤記修正 NWDGM114 誤記修正 NWDGM115 誤記修正	NCES
1.0.3	2017/2/16	NWDGM133 追加	NCES
1.0.4	2017/3/6	リファレンス 仕様番号一覧更新	NCES