ADAPTIVE CRUISE CONTROL

software requirement & Design Spec.

自適應巡航控制

軟體需求及設計規格

ACC-srds

Version-00.01

|  |  |  |  |
| --- | --- | --- | --- |
| **Authors** | Fenix Tsai | **Date** | 2022/09/16 |
| Contributors | Fenix Tsai | **Date** |  |
| Approved | *DRAFT* | **Date** |  |

校訂記錄

|  |  |  |  |
| --- | --- | --- | --- |
| **REVISION** | **DATE** | **AUTHOR** | **SECTIONS/PAGES AFFECTED** |
| **REMARKS** | | |
| 00.01 | 2022.09.02 | Fenix Tsai |  |
| 初版 – 文件建立 | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |
|  |  |  |  |
|  | | |

目錄

[1 簡介 4](#_Toc113962173)

[1.1 說明 4](#_Toc113962174)

[1.2 閱讀權限 4](#_Toc113962175)

[1.3 縮寫／術語定義 4](#_Toc113962176)

[1.3.1 縮寫定義 4](#_Toc113962177)

[1.3.2 名詞定義 4](#_Toc113962178)

[1.4 編號格式原則 5](#_Toc113962179)

[1.5 參考文獻 5](#_Toc113962180)

[2 模組功能需求 6](#_Toc113962181)

[2.1 模組功能彙整 6](#_Toc113962182)

[3 模組需求及邏輯設計 7](#_Toc113962183)

[3.1 ACC\_SRDS\_01 ACC狀態決策 8](#_Toc113962184)

[3.1.1 ACC\_SRDS\_01\_001 ACC系統狀態邏輯設計 9](#_Toc113962185)

[3.1.2 ACC\_SRDS\_01\_002 Disable狀態執行流程 11](#_Toc113962186)

[3.1.3 ACC\_SRDS\_01\_003 Enable狀態執行流程 12](#_Toc113962187)

[3.1.4 ACC\_SRDS\_01\_004 Standby狀態執行流程 13](#_Toc113962188)

[3.1.5 ACC\_SRDS\_01\_005 Active狀態執行流程 14](#_Toc113962189)

[3.1.6 ACC\_SRDS\_01\_006 Active／初始化執行流程 16](#_Toc113962190)

[3.1.7 ACC\_SRDS\_01\_007 Active／速度模式狀態執行流程 17](#_Toc113962191)

[3.1.8 ACC\_SRDS\_01\_008 Active／維持模式狀態執行流程 18](#_Toc113962192)

[3.1.9 ACC\_SRDS\_01\_009 Active／停頓模式狀態執行流程 19](#_Toc113962193)

[3.1.10 ACC\_SRDS\_01\_010 Active／覆寫模式狀態執行流程 20](#_Toc113962194)

[3.2 ACC\_SRDS\_02 人機介面邏輯決策 21](#_Toc113962195)

[3.2.1 ADCU\_SRDS\_02\_001 系統跛行模式判斷 22](#_Toc113962196)

[Appendix A 24](#_Toc113962197)

# 簡介

## 說明

此文件為ADCU中ACC模組之軟體需求及設計規格文件。主要是依據軟體需求矩陣文件SWRM、SWAC及IOIF文件中提出所需的輸入輸出訊號，來彙整此模組功能需求，並依據需求來設計適當的軟體邏輯來對應。

## 閱讀權限

本文僅供 CubTEK跟**NTUT**內部的工程人員和管理人員使用。未經另一方書面許可，不得在這些組織之外散佈。

## 縮寫／術語定義

### 縮寫定義

|  |  |
| --- | --- |
| **名詞** | **描述** |
| ADCU | ADAS Control Unit |
| ACC | Adaptive Cruise Control |
| SWRM | Software Requirement Matrix |
| SWAC | Software Architecture Chart |
| IOIF | Input /Output signal Information |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

### 名詞定義

1. XXX：TBD

## 編號格式原則

本文中將需求及設計定義成一種編號，這些ID應具有唯一性，此 ID由固定字串和數字組成：

軟體需求ID：XXXX\_SRDS\_YY\_ZZ

XXXX：模組名稱縮寫

YY：功能需求第一階層編號

ZZ：功能需求第二階層編號

Ex ：（ACC\_SRDS\_01 or ACC\_SRDS\_01\_001）

## 參考文獻

1. TBD
2. TBD

# 模組功能需求

此章節說明由SWRM文件中所分析出用於ACC模組的功能需求

## 模組功能彙整

Table 2-1 ACC模組功能需求表

|  |  |
| --- | --- |
| SWRM編號 | 需求描述 |
| ADCU\_SWRM\_001 |  |
| ADCU\_SWRM\_002 |  |
|  |  |
|  |  |
|  |  |
|  |  |

# 模組需求及邏輯設計

此章節描述該模組之詳細功能需求描述，並依據描述設計邏輯及軟體搭建方式

## ACC\_SRDS\_01 ACC模組架構

此模組底下將區分成四個子系統來運作，分別為ACC系統狀態決策／ACC控制狀態決策／ACC驅動控制計算／ACC剎車控制計算，如下圖所示：



Figure 3-1 模組架構圖

其中需當系統狀態決策判定ACC為正作動狀態時，才允許執行其他三個子系統功能，控制順序及邏輯如下圖所示：



Figure 3-2 各子功能執行順序

當ACC系統控制狀態決策輸出VACC\_ACCActing\_flg == TRUE時，ACC控制狀態決策／ACC驅動控制計算／ACC剎車控制計算此三個子系統才可允許啟動計算邏輯。

## ACC\_SRDS\_02 ACC系統狀態決策

此子系統功能主要用來呈現當下ACC功能所需運行的狀態，而輸入訊號分別為駕駛人控制需求／駕駛者介入行為／動力、剎車系統，及感知器接收狀態，並根據這些訊號、狀態來進行ACC系統狀態的轉換判斷。

此子功能之訊號I/O如下所示：

**輸入訊號**：

* VIFO\_ACCTurnOnReq\_flg
* VIFO\_ACCTurnOffReq\_flg
* VIFO\_ACCVehClearance\_m
* VJUD\_ACCEnableReq\_flg
* VJUD\_ACCActiveReq\_flg
* VJUD\_ACCDeactivationReq\_flg
* VJUD\_ACCStandstillReq\_flg
* VJUD\_ACCRefollowReq\_flg
* VJUD\_ACCOverrideReq\_flg
* VACC\_ACCHoldReqEnablePrev\_flg
* KACC\_DefStandstillClearance\_m

**輸出訊號**：

* VACC\_ACCStateHMI\_enum
* VACC\_ACCSystemState\_enum
* VACC\_ACCTqReqAppl\_flg
* VACC\_ACCDecelReqAppl\_flg
* VACC\_ACCHoldReq\_flg
* VACC\_StandstillClearance\_m
* VACC\_ACCActing\_flg
* VACC\_ACCOverride\_flg

### ACC\_SRDS\_02\_001 ACC系統狀態邏輯設計

根據SWRM內容，設計一ACC系統狀態主架構流程，於下圖所示：



Figure 3-3狀態決策主架構流程圖

1. 狀態定義：VACC\_ACCSystemState\_enum

* Disable狀態 ： ENUM\_ACCSTATE\_DISABLED = 0
* Enable狀態 ： ENUM\_ACCSTATE\_ENABLE = 1
* Standby狀態 ： ENUM\_ACCSTATE\_STANDBY = 2
* Active狀態 ： 於Active狀態執行流程中定義

1. Disable狀態🡪Enable狀態之條件判斷：

* VJUD\_ACCEnableReq\_flg == TRUE

1. Enable狀態🡪Standby狀態之條件判斷

* VIFO\_ACCTurnOnReq\_flg == TRUE

1. Enable狀態🡪Disable狀態之條件判斷

* VJUD\_ACCEnableReq\_flg == FALSE

1. Standby狀態🡪Active狀態之條件判斷

* VJUD\_ACCActiveReq\_flg == TRUE

1. Standby狀態🡪Enable狀態之條件判斷

* VIFO\_ACCTurnOffReq\_flg == FALSE

1. Standby狀態🡪 Disable狀態之條件判斷

* VJUD\_ACCEnableReq\_flg == FALSE

1. Active狀態🡪 Standby狀態之條件判斷

* VJUD\_ACCDeactivationReq\_flg == TRUE

1. Active狀態🡪 Disable狀態之條件判斷

* VJUD\_ACCEnableReq\_flg == FALSE

### ACC\_SRDS\_02\_002 Disable狀態執行流程



Figure 3-4 Disable狀態流程圖

在此狀態中，將依序針對以下訊號／狀態進行初始化：

1. VACC\_ACCSystemState\_enum = ENUM\_ACCSTATE\_DISABLED
2. VACC\_ACCStateHMI\_enum = ENUM\_ACCHMISTATE\_ERROR
3. VACC\_ACCTqReqAppl\_flg = FALSE
4. VACC\_ACCDecelReqAppl\_flg = FALSE
5. VACC\_ACCActing\_flg = FALSE
6. VACC\_ACCHoldReq\_flg = FALSE
7. VACC\_ACCOverride\_flg = FALSE
8. VACC\_StandstillClearance\_m = KACC\_DefStandstillClearance\_m

### ACC\_SRDS\_02\_003 Enable狀態執行流程



Figure 3-5 Enable狀態流程圖

在此狀態中，將依序針對以下訊號／狀態進行初始化：

1. VACC\_ACCSystemState\_enum = ENUM\_ACCSTATE\_ENABLED
2. VACC\_ACCStateHMI\_enum = ENUM\_ACCHMISTATE\_OFF

### ACC\_SRDS\_02\_004 Standby狀態執行流程



Figure 3-6 Enable狀態流程圖

在此狀態中，將依序針對以下訊號／狀態進行初始化：

1. VACC\_ACCSystemState\_enum = ENUM\_ACCSTATE\_STANDBY
2. VACC\_ACCStateHMI\_enum = ENUM\_ACCHMISTATE\_STANDBY
3. VACC\_ACCActing\_flg = FALSE
4. VACC\_ACCTqReqAppl\_flg=FALSE
5. VACC\_ACCDecelReqAppl\_flg=FALSE
6. VACC\_ACCHoldReq\_flg = FALSE

### ACC\_SRDS\_02\_005 Active狀態執行流程



Figure 3-7 Active狀態流程圖

1. 狀態定義：VACC\_ACCSystemState\_enum

* 速度模式狀態 ： ENUM\_ACCSTATE\_SPEED = 3
* 維持模式狀態 ： ENUM\_ACCSTATE\_HOLD = 4
* 停頓模式狀態 ： ENUM\_ACCSTATE\_STANDSTILL = 5
* 覆寫模式狀態 ： ENUM\_ACCSTATE\_ OVERRIDE = 6

1. 速度模式狀態🡪維持模式狀態之條件判斷

* VACC\_ACCHoldReqEnablePrev\_flg == TRUE

1. 維持模式狀態🡪速度模式狀態之條件判斷

* VACC\_ACCHoldReqDisablePrev\_flg == TRUE

1. 維持模式狀態🡪停頓模式狀態之條件判斷

* VJUD\_ACCStandstillReq\_flg == FALSE

1. 停頓模式狀態🡪速度模式狀態之條件判斷

* VJUD\_ACCRefollowReq\_flg == TRUE

1. 速度模式狀態🡪 覆寫模式狀態之條件判斷

* VJUD\_ACCOverrideReq\_flg == TRUE

1. 覆寫模式狀態🡪 速度模式狀態之條件判斷

* VJUD\_ACCOverrideReq\_flg == FALSE

### ACC\_SRDS\_02\_006 Active／初始化執行流程



Figure 3-8 Active／初始化流程圖

在此狀態中，將依序針對以下訊號／狀態進行初始化：

1. VVACC\_ACCStateHMI\_enum = ENUM\_ACCHMISTATE\_ACTIVE
2. VACC\_ACCActing\_flg = TRUE
3. VACC\_ACCTqReqAppl\_flg = TRUE
4. VACC\_ACCDecelReqAppl\_flg = TRUE

### ACC\_SRDS\_02\_007 Active／速度模式狀態執行流程



Figure 3-9 Active／速度模式狀態流程圖

在此狀態中，將依序針對以下訊號／狀態進行初始化：

1. VACC\_ACCSystemState\_enum = ENUM\_ACCSTATE\_SPEED;
2. VACC\_ACCHoldReq\_flg = FALSE;
3. VACC\_StandstillClearance\_m = KACC\_DefStandstillClearance\_m;
4. VACC\_ACCOverride\_flg = FALSE;
5. VACC\_ACCTqReqAppl\_flg = TRUE;
6. VACC\_ACCDecelReqAppl\_flg = TRUE;

### ACC\_SRDS\_02\_008 Active／維持模式狀態執行流程



Figure 3-10 Active／維持模式狀態流程圖

在此狀態中，將依序針對以下訊號／狀態進行初始化：

1. VACC\_ACCSystemState\_enum = ENUM\_ACCSTATE\_HOLD
2. VACC\_ACCHoldReq\_flg= TRUE

### ACC\_SRDS\_02\_009 Active／停頓模式狀態執行流程



Figure 3-11 Active／停頓模式狀態流程圖

在此狀態中，將依序針對以下訊號／狀態進行初始化：

1. VACC\_ACCSystemState\_enum = ENUM\_ACCSTATE\_STANDSTILL
2. VACC\_ACCTqReqAppl\_flg = FALSE
3. VACC\_StandstillClearance\_m = VIFO\_ACCVehClearance\_m

### ACC\_SRDS\_02\_010 Active／覆寫模式狀態執行流程



Figure 3-12 Active／覆寫模式狀態流程圖

在此狀態中，將依序針對以下訊號／狀態進行初始化：

1. VACC\_ACCSystemState\_enum = ENUM\_ACCSTATE\_OVERRIDE;
2. VACC\_ACCOverride\_flg=TRUE;
3. VACC\_ACCTqReqAppl\_flg=FALSE;
4. VACC\_ACCDecelReqAppl\_flg=FALSE;

## ACC\_SRDS\_03 ACC控制狀態決策

此子系統接收本車資訊／目標車的相對資訊／駕駛者對於系統的設定狀態，來計算ACC系統中的控制需求狀態是否成立，此子系統底下會再分為三個模組處理，如下圖所示：



Figure 3-13 ACC控制狀態決策架構圖

此子功能之訊號I/O如下所示：

**輸入訊號**：

* VDINR\_VehSpd\_mps
* VJUD\_ACCSetSpeed\_kph
* VJUD\_ACCSetTimeGap\_s
* VIFO\_ACCTargetVehRelSpeedX\_mps
* VACC\_ACCTargetVehSpd\_mps
* VIFO\_ACCTGap\_s
* VIFO\_ACCTargetVehAssigned\_flg
* VIFO\_ACCDist\_flg
* VACC\_ACCOverride\_flg
* VIFO\_ACCMCLongAcc\_mps2
* VIFO\_ACCVehClearance\_m

**輸出訊號**：

* VACC\_TorqueReleaseReq\_flg
* VACC\_ACCRefSpd\_mps
* VACC\_DecelControlReq\_flg
* VACC\_DesiredDistance\_m
* VACC\_ACCDistControlSel\_flg
* VACC\_ACCBrakeToStop\_flg
* VACC\_ACCMCBraking\_flg
* VACC\_ACCMCDecel\_mps2
* VACC\_ACCBrakeReleaseSituation\_flg

### ACC\_SRDS\_03\_001 參考車速計算

此子系統用於計算後續控制驅動／剎車時的參考車速，使用合理（最小）兩車距離、駕駛人設定的時間間距及駕駛人設定車速，前兩者使用PID控制來分別計算出參考車速值，與最後者根據在不同車速的條件下，來決定使用哪一個參考車速。

依序執行以下策略：

1. 當前車速與設定之時間間格計算出距離前車值，與最小安全車距值作比較取最大值後，作為與目標車期望間距值，其邏輯設計如下：

* VACC\_MinDesiredDistance\_m = KACC\_OffsetToBumper\_m + KACC\_StoppingDistSafety\_m，其中KACC\_OffsetToBumper\_m = 3和KACC\_StoppingDistSafety\_m = 5
* **VACC\_DesiredDistance\_m** = Max（VACC\_MinDesiredDistance\_m, VDINR\_VehSpd\_mps \*VJUD\_ACCSetTimeGap\_s）

1. 以與前車期望間距值做為控制目標，而偵測與前車間距為回饋值，並以當前車速作為增益值計算條件，使用PID控制（微分器與積分器需預設為關閉，但可用調校值開啟）來計算車輛加減速控制訊號，其邏輯設計如下：

* TargetValue = VACC\_DesiredDistance\_m
* FeedbackValue = VIFO\_ACCVehClearance\_m
* FeedForward = KACC\_ACCDistFFParam\_num = 0
* Error\_Gain = VDINR\_VehSpd\_mps 轉換單位為kph後，帶入1-D table（AACC\_ACCDistVehSpd\_kph , MACC\_DistPIDErrorG\_num）

其中AACC\_ACCDistVehSpd\_kph = [0 5 10 15 20 25 30 35 40 45 50 55]

其中MACC\_DistPIDErrorG\_num = [-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1]

* P\_Gain = VDINR\_VehSpd\_mps 轉換單位為kph後，帶入1-D table（AACC\_ACCDistVehSpd\_kph , MACC\_DistPIDErrorG\_num）

其中AACC\_ACCDistVehSpd\_kph = [0 5 10 15 20 25 30 35 40 45 50 55]

其中MACC\_DistPIDKp\_num = 0.4 0.4 0.4 0.65 0.65 0.65 0.65]

* I\_Gain = KACC\_ACCDistKiParam\_num = 0
* D\_Gain = KACC\_ACCDistKdParam\_num = 0
* ITerm\_Initial = KACC\_ACCDistInitialIParam\_num = 0
* ITerm\_min = KACC\_ACCDistMinIParam\_num = 0
* ITerm\_max = KACC\_ACCDistMaxIParam\_num = 0
* ITerm\_reset = KACC\_ACCDistResetIParam\_num = 0
* ControlHold = FALSE = 0
* 此PID控制模組僅輸出**VACC\_FollowModeDistSpdPID\_mps**，其餘訊號則需terminate

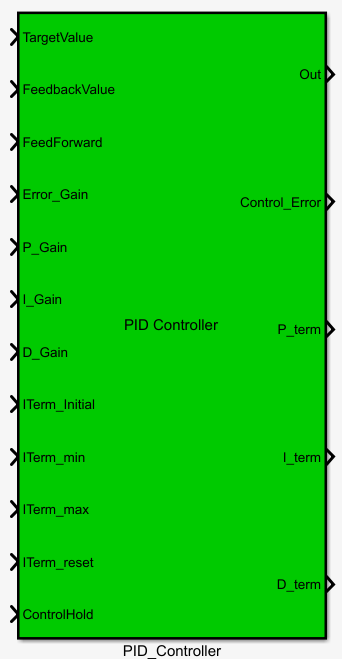


Figure 3-14 PID模組圖

1. 以仲裁完之ACC設定時間間距狀態做為控制目標，而ACC兩車時間間距為回饋值，並以當前仲裁完之ACC設定時間間距狀態／ACC兩車時間間距／ACC兩車相對距離／ACC目標物相對車速／診斷後之車速1訊號作為增益值計算條件，使用PID控制（微分器與積分器需預設為關閉，但可用調校值開啟）來計算車輛加減速控制訊號，其邏輯設計如下：

* TargetValue = VJUD\_ACCSetTimeGap\_s
* FeedbackValue = VIFO\_ACCTGap\_s
* FeedForward = KACC\_ACCGapFFParam\_num = 0
* Error\_Gain = KACC\_ACCGapErrorGParam\_num = 0
* P\_Gain = 使用一狀態機圖，並根據不同條件來計算出不同值，邏輯設計如下：
* VACC\_ACCRecoverStatKp\_flg（狀態輸入訊號1）= （VIFO\_ACCTargetVehRelSpeedX\_mps > KACC\_ACCRelSpdKpTrans\_kph\*kph2mps ）OR （VIFO\_ACCTGap\_s < (VJUD\_ACCSetTimeGap\_s + KACC\_ACCTGapTransMargin\_s)）

其中KACC\_ACCRelSpdKpTrans\_kph = 1

其中KACC\_ACCTGapTransMargin\_s = 0.1

* VACC\_TimeGapChange\_flg（狀態輸入訊號2）= （ VJUD\_ACCSetTimeGap\_s(Z) ≠ VJUD\_ACCSetTimeGap\_s(Z-1) ）
* VACC\_ACCDistEnable\_flg（狀態輸入訊號3） = （ VJUD\_ACCSetTimeGap\_s ≠ KJUD\_ACCMinTimeGap\_s ） AND VIFO\_ACCDist\_flg
* VACC\_ACCGapKp\_num（狀態輸入訊號4） = VDINR\_VehSpd\_mps 轉換單位為kph後，帶入1-D table（AACC\_VehSpdPID\_kph , MACC\_TGapRefSpdKp\_num）

其中AACC\_VehSpdPID\_kph =

[0 10 20 30 40 50 60 70 80 90 100 110 120 130 140]

其中MACC\_TGapRefSpdKp\_num =

[2 2 4 5 5 7 10 17 20 24 30 35 38 38 38 38]

* KACC\_ACCGapKpParamTrans\_num（狀態輸入訊號5） = 0
* 狀態機條件判斷及流程如下圖所示，最終輸出VACC\_ACCGapKpAppl\_num為PID之P Gain值：



Figure 3-15 TGAP P增益值判斷流程圖

* I\_Gain = KACC\_ACCGapKiParam\_num = 0
* D\_Gain = KACC\_ACCGapKdParam\_num = 0
* ITerm\_Initial = KACC\_ACCGapInitialIParam\_num = 0
* ITerm\_min = KACC\_ACCGapMinIParam\_num = -100
* ITerm\_max = KACC\_ACCGapMaxIParam\_num = 100
* ITerm\_reset = KACC\_ACCGapResetIParam\_num = 0
* ControlHold = FALSE = 0
* 此PID控制模組僅輸出**VACC\_FollowModeGapSpdPID\_mps**，其餘訊號則需terminate

1. 根據當前車速來決定切換**VACC\_FollowModeGapSpdPID\_mps**或**VACC\_FollowModeDistSpdPID\_mps**，將其值用來補償車速並最終作為維持模式時的設定車速值**VACC\_FollowModeSetSpd\_mps**，其邏輯設計如下：

* 當VDINR\_VehSpd\_mps ≧ KACC\_DistControlEnableSpd\_kph（使用Hysteresis），令VACC\_ACCTGapControlSel\_flg == TRUE

其中KACC\_DistControlEnableSpd\_kph = 49

* 當VDINR\_VehSpd\_mps < KACC\_DistControlDisableSpd\_kph（使用Hysteresis），令VACC\_ACCTGapControlSel\_flg == FALSE

其中KACC\_DistControlDisableSpd\_kph = 4

* VACC\_ACCDistControlSel\_flg = NOT（VACC\_ACCTGapControlSel\_flg）
* 如 VACC\_ACCDistControlSel\_flg = TRUE，則VACC\_RawFollowModeSetSpd\_mps= （VACC\_FollowModeDistSpdPID\_mps + VDINR\_VehSpd\_mps）
* 如 VACC\_ACCDistControlSel\_flg = FALSE，則VACC\_RawFollowModeSetSpd\_mps= （VACC\_FollowModeGapSpdPID\_mps + VDINR\_VehSpd\_mps）
* **VACC\_FollowModeSetSpd\_mps** = MAX（

MIN（VACC\_RawFollowModeSetSpd\_mps, KACC\_RawFMUpperLimit\_mps）, KACC\_RawFMLowerLimit\_mps）

其中KACC\_RawFMUpperLimit\_mps = 50

其中KACC\_RawFMLowerLimit\_mps = 0

1. 當以下條件任一成立時，則令仲裁完之ACC設定目標車速狀態為ACC參考車速，如無成立，則令最小跟隨模式車速為ACC參考車速：

條件一、仲裁完之ACC設定目標車速狀態低於目標車之車速

條件二、目標車無視別狀態為成立

條件三、仲裁完之ACC設定目標車速狀態低於最小跟隨模式車速

* 如（（VACC\_ACCSetSpeed\_mps < VACC\_ACCTargetVehSpd\_mps）OR（VACC\_ACCTargetVehNotAssigned\_flg = TRUE）OR（VACC\_ACCSetSpeed\_mps < VACC\_MinFollowModeSpd\_mps）） = TRUE，則令NACC\_ACCRefSpd\_mps = VACC\_ACCSetSpeed\_mps，反之則令NACC\_ACCRefSpd\_mps = VACC\_MinFollowModeSpd\_mps

1. 使用移動平均公式將計算出來的＂跟隨模式下最小控制速度＂進行濾波，消除短期的波動現象，達到平順化效果，邏輯設計如下：

* 下圖其一為移動平均公式訊號I/O圖，而移動平均之使用公式為：

Equation 3-1



Figure 3-16 移動平均架構圖

* New\_In\_Value = NACC\_ACCRefSpd\_mps
* New\_Ave\_Value = VACC\_ACCRefSpd\_mps
* Old\_Ave\_Value = VACC\_ACCRefSpd\_mps（t-1）
* Ts = C\_TICK\_TIME\_S = 0.005
* Tc = VACC\_ACCRefSpdRollAvgInt\_s，其計算將使用一狀態機流程，並針對不同的條件下，決定適合的值，邏輯設計如下
* 輸入訊號1：NACC\_ACCRefSpdRollAvgSwitch\_flg 之值由以下判斷式來計算，（（NACC\_ACCRefSpd\_mps(t)- NACC\_ACCRefSpd\_mps(t-1)）> kph2mps\* KACC\_ACCRefSpdRollAvgMargin\_kph）

其中KACC\_ACCRefSpdRollAvgMargin\_kph = 1.5

* 輸入訊號2：KACC\_ACCRefSpdShortInt\_s = 0.5
* 輸入訊號3：KACC\_ACCRefSpdLongInt\_s = 1
* 輸入訊號4：VACC\_ACCRefSpdTimer\_stp = KACC\_ACCRefSpdTimer\_s / C\_TICK\_TIME\_S

其中KACC\_ACCRefSpdTimer\_s = 1

其中 C\_TICK\_TIME\_S = 0.005

* 輸出訊號1：VACC\_ACCRefSpdRollAvgInt\_s，根據下圖狀態機流程進行判斷及切換數值 ：



Figure 3-17 時間常數計算流程圖

* Aa

### ACC\_SRDS\_03\_002 目標車狀態判斷

TBD。

依序執行以下策略：

1. 利用移動平均公式將計算出來的＂ACC 最關鍵車輛的縱向加速度＂訊號進行濾波，消除短期的波動現象，達到平順化效果，邏輯設計如下：

* 下圖其一為移動平均公式訊號I/O圖，而移動平均之使用公式為：

Equation 3-2



Figure 3-17 移動平均架構圖

* New\_In\_Value = VIFO\_ACCMCLongAcc\_mps2
* New\_Ave\_Value = **VACC\_ACCMCLongAcc\_mps2**
* Old\_Ave\_Value = VACC\_ACCMCLongAcc\_mps2（t-1）
* Ts = C\_TICK\_TIME\_S = 0.005
* Tc = KACC\_MCLongAccelRollAvgInterval\_s = 0.5

1. ＂關鍵車輛正在剎車的狀態＂是在ACC距離控制啟用後，以ACC 最關鍵車輛的縱向加速度在不同門檻值下，來判斷是否成立：

* 當**VACC\_ACCMCLongAcc\_mps2** > Condition1（使用Hysteresis），令VACC\_ACCMCBraking\_flg == TRUE
* Condition1 = KACC\_ACCDistCtrlTargetDecelThres\_mps2 =，當VACC\_ACCDistControlSel\_flg = TRUE
* Condition1 = KACC\_ACCTargetDecelThres\_mps2 =，當VACC\_ACCDistControlSel\_flg = FALSE
* 其中KACC\_ACCDistCtrlTargetDecelThres\_mps2 = 0.25
* 其中KACC\_ACCTargetDecelThres\_mps2 = 0.3
* 當**VACC\_ACCMCLongAcc\_mps2** < Condition1（使用Hysteresis），令VACC\_ACCMCBraking\_flg == FALSE
* 其中KACC\_ACCTargetDecelOffThres\_mps2 = 0.1

1. ＂ACC正處於暫態＂是以Time-Gap控制暫態啟動狀態／距離控制暫態啟動狀態／目標車已是別狀態來決定是否成立：VACC\_ACCTransitory\_flg計算：

* 當VACC\_ACCDistControlSel\_flg = TRUE，會開啟距離控制暫態啟動狀態的判斷程序，否則則開啟Time-Gap控制暫態啟動狀態的判斷程序
* 距離控制暫態啟動狀態 = NACC\_DistTransitoryEnable\_flg，將由以下條件式否任一來判斷是否成立：
* VACC\_ACCMCBraking\_flg
* VACC\_TransDistApproachSlowTarget\_flg ，其值為AND（VACC\_SlowTargetDistApproach\_flg = （VIFO\_ACCTargetVehRelSpeedX\_mps < kph2mps\*KACC\_TransitoryRelSpd\_kph）, VACC\_DistControlApproach\_flg = （VIFO\_ACCVehClearance\_m > （VACC\_DesiredDistance\_m + KACC\_TransitoryDesDistApproach\_m）））

其中KACC\_TransitoryRelSpd\_kph = -5

其中KACC\_TransitoryDesDistApproach\_m = 0.5

* VACC\_TransDistTargetCutIn\_flg，其值為AND（VACC\_SlowTargetDistApproach\_flg, （（VIFO\_ACCVehClearance\_m（t-1）- VIFO\_ACCVehClearance\_m（t））> KACC\_TransitoryClearanceDec\_m））

其中KACC\_TransitoryClearanceDec\_m = 2

* VACC\_TransDistIncrease\_flg，其值為（VACC\_DesiredDistance\_m（t）- VACC\_DesiredDistance\_m（t-1））> KACC\_TransitoryClearanceDec\_m）

其中KACC\_TransDistIncMargin\_m = 2

* VACC\_TransDistIntrusion\_flg，其值為（VIFO\_ACCVehClearance\_m < （VACC\_DesiredDistance\_m - KACC\_TransitoryDistIntrusion\_m））

其中KACC\_TransitoryDistIntrusion\_m = 2.5

* Time-Gap控制暫態啟動狀態 = NACC\_TGapTransitoryEnable\_flg，將由以下條件式否任一來判斷是否成立：
* VACC\_ACCMCBraking\_flg
* VACC\_TransApproachSlowTarget\_flg，其值為AND（VACC\_SlowTargetApproach\_flg = （VIFO\_ACCTargetVehRelSpeedX\_mps < kph2mps\*KACC\_TransitoryRelSpd\_kph）, VACC\_TimeGapApproach\_flg =（ VIFO\_ACCTGap\_s > （VJUD\_ACCSetTimeGap\_s + KACC\_TransitorySetTGapApproach\_s）））

其中KACC\_TransitorySetTGapApproach\_s = 0.1

* VACC\_TransTargetCutIn\_flg，其值為AND（（（VIFO\_ACCTGap\_s（t-1） - VIFO\_ACCTGap\_s（t）） > KACC\_TransitoryTGapDec\_s）, VACC\_SlowTargetApproach\_flg）

其中KACC\_TransitoryTGapDec\_s = 0.2

* VACC\_TransSetTGapIncrease\_flg，其值為（VJUD\_ACCSetTimeGap\_s（t）- VJUD\_ACCSetTimeGap\_s（t-1）） > 0
* VACC\_TransTGapIntrusion\_flg，其值為VJUD\_ACCSetTimeGap\_s <（VJUD\_ACCSetTimeGap\_s - KACC\_TransitorySetTGapIntrusion\_s）

其中KACC\_TransitorySetTGapIntrusion\_s = 0.3

* Time-Gap控制暫態啟動狀態及距離控制暫態啟動狀態的判斷結果將耦合在一起
* ACC正處於暫態 = **VACC\_ACCTransitory\_flg** ，其值為AND（上述耦合結果, VIFO\_ACCTargetVehAssigned\_flg）

1. ACC剎車致車輛停止狀態 = **VACC\_ACCBrakeToStop\_flg**，其值為以下條件任一成立時狀態成立：

* Condition1 = AND（VACC\_ACCMCBraking\_flg, （VACC\_ACCTargetVehSpd\_mps ≦ kph2mps\* KACC\_BTSTargetBrakingSpd\_kph））

其中KACC\_BTSTargetBrakingSpd\_kph = 10

* Condition2 = VACC\_ACCTargetVehSpd\_mps ≦ kph2mps\* KACC\_BTSTargetNoBrakingSpd\_kph

其中KACC\_BTSTargetNoBrakingSpd\_kph = 3

### ACC\_SRDS\_03\_003 減速控制狀態判斷

TBD。

Appendix A

無。