Date Standard Profess						Detaile	d API Desci	ription				
Date_Handler	ECU name	SWC name	API Description	Scope	Type				Input Parameters	Parameter Description	Return Parameters	Parameter Description
Description			main()	Private	Function	Non-reentrant	Syncronus	Non-Recursion	void	NA	void	NA
Column C		Data_Handler	Data_Handler_Init()	public	Function	Non-reentrant	Syncronus	Non-Recursion	void	NA	void	NA
PR. Stat			Data_Update()	public	Function	Non-reentrant	Syncronus	Non-Recursion	void	NA	void	NA
Processor Proc			Data_Send()	public	Function	Reentrant	Syncronus	Non-Recursion	void	NA	void	NA
Fig. Carbinat Public P		DR_Stat	DR_Stat_Init	public	Function	Non-reentrant	Syncronus	Non-Recursion	* ConfigPtr		void	NA
DR. Quellate			DR_UpdateStat	public	Function	Non-reentrant	Syncronus	Non-Recursion	DR_ld	Identify required Door Id	void	NA
SW_State			DR_GetStat	public	Function	Reentrant	Syncronus	Non-Recursion	DR_ld		DR_StatType	return Job Status (OK in cause or success, NOK in case of Failure)
SW_State		SW_Stat	SW_Stat_Init	public	Function	Non-reentrant	Syncronus	Non-Recursion	* ConfigPtr		void	NA
SPU_State			SW_UpdateStat	public	Function	Non-reentrant	Syncronus	Non-Recursion	SW_ld	Identify required SW Id	void	
SPD_State public Function Non-receitant Syncronus Non-Recursion SPD_Id Identify required SPD centrol of void Non-Recursion SPD_Id Identify required SPD centrol of Void Non-Recursion SPD_Id Identify required SPD centrol of SPD_Staffyge return into SMA_CVC Non-Recursion Void Non-Recursion Void	ECU 1		SW_GetStat	public	Function	Reentrant	Syncronus	Non-Recursion	SW_ld		SW_StatType	return Job Status (OK in cause or success, NOK in case of Failure)
SPD_State SPD_GelStat public Function Reentrant Syncronus Non-Recursion SPD_I Identify required SPD sentant Id SPD_Staff; per success, Notice SPD_Interfered SPD_state (not success, Notice SPD_State (n			SPD_Stat_Init	public	Function	Non-reentrant	Syncronus	Non-Recursion	* ConfigPtr		void	NA
SPU_State			SPD_UpdateStat	public	Function	Non-reentrant	Syncronus	Non-Recursion	SPD_ld	Identify required SPD sensor Id	void	NA
SPD_TickHander		SPD_Stat	SPD_GetStat	public	Function	Reentrant	Syncronus	Non-Recursion	SPD_ld	Identify required SPD sensor Id	SPD_StatType	return Job Status (OK in cause or success, NOK in case of Failure)
TMR Int			SPD_TimerHandler	public	ISR	Non-reentrant	Syncronus	Non-Recursion	void	NA	void	NA
TMR_SharTimer public Function Repertant Syncronus Non-Recursion Non-Recursion ColingPut Configured data Void NA			SPD_TickHandler	public	ISR	Non-reentrant	Syncronus	Non-Recursion	void		void	NA
MM_Start inner		TMR	TMR_Init	public	Function	Non-reentrant	Syncronus	Non-Recursion	* ConfigPtr		void	NA
TMR_Stortmer			TMR_StartTimer	public	Function	Reentrant	Syncronus	Non-Recursion		-	void	NA
SW_StateHandler SW_StateHandler_init			TMR StopTimer	public	Function	Reentrant	Syncronus	Non-Recursion			void	NA NA
SW_StateHandler_Initt			- '	•	+	1				,		
SW_State-hander_StM public Function Non-reentrant Syncronus Non-Recursion void NA void NA		SW StateHandler	SW StateHandler Initt		+		_	.				
LED_Ctrl Int Dublic Function Non-reentrant Syncronus Non-Recursion *ConfigPt Pointer to Driver Pre-Compiler configured data void NA		SW_Statemanuler					_	-				
LED_Cital LED_StatUpdate public Function Non-reentrant Syncronus Non-Recursion Syncronus Syncronus		LED_Ctrl					-			Pointer to Driver Pre-Compiler		
BZR_Stat_Init public Function Non-reentrant Syncronus Non-Recursion ConfigPtr Pointer to Driver Pre-Compiler configured data Void NA									LED Id		void	NA
BZR_Stat_Init public Function Non-reentrant Syncronus Non-Recursion ConfigPt Config			LED_StatUpdate	public	Function	Non-reentrant	Syncronus	Non-Recursion	state	Define required LED State	void	NA
BZR_UpdateStat public Function Non-reentrant Syncronus Non-Recursion 12R_Stat Define required BZR State to be setted 9 void NA NA NA NA Std_Type Function Non-reentrant Syncronus Non-Recursion 12R_State Define required BZR State to be setted 9 void NA NA NA Std_Type Function Reentrant Syncronus Non-Recursion Channelld Identify controlled PVM channel Id void NA NA NA Std_Type Function Reentrant Syncronus Non-Recursion Channelld Identify controlled PVM channel Id void NA			BZR_Stat_Init	public	Function	Non-reentrant	T .				void	NA
PWM PWM_Start public Function Non-recentrant Syncronus Non-Recursion EZR_Stat Define required BZR State to be setted		BZR_Ctrl					Syncronus	Non-Recursion	BZR_ld	· · ·	void	NA
PWM Init public Function Non-reentrant Syncronus Non-Recursion Configured data void NA PWM_Start public Function Reentrant Syncronus Non-Recursion Channelld Identify controlled PVM channel Id void NA BCM_Handler_Init public Function Reentrant Syncronus Non-Recursion Channelld Identify controlled PVM channel Id void NA NA Std_Type Function NA PWM_Stop public Function Non-reentrant Syncronus Non-Recursion void NA BCM_Handler_Init public Function Reentrant Syncronus Non-Recursion void NA BCM_Handler_Init public Function Reentrant Syncronus Non-Recursion void NA BCM_Handler PWM_State Public Function Reentrant Syncronus Non-Recursion void NA BCM_Handler PWM_State Public Function Reentrant Syncronus Non-Recursion void NA BCM_Handler PWM_State Public Function Reentrant Syncronus Non-Recursion void NA BCM_Handler PWM_State Public Function Reentrant Syncronus Non-Recursion void NA BCM_Handler PWM_State Public Function Non-reentrant Syncronus Non-Recursion void NA BCM_Handler PWM_State Public Function Non-reentrant Syncronus Non-Recursion void NA Std_Type return_Job Status (OK or success, NOK in Failure) BCMH_Update_SM_State Public Function Non-reentrant Syncronus Non-Recursion void NA Std_Type return_Job Status (OK or success, NOK in Failure) BCMH_Update_SPD_State Public Function Non-reentrant Syncronus Non-Recursion void NA BCM_Hupdate_SPD_State Public Function Non-reentrant Syncronus Non-Recursion void NA BCM_Hupdat			BZR_UpdateStat	public	Function	Non-reentrant			BZR_Stat	-		
PWM_Start		PWM	PWM_Init	public	Function	Non-reentrant	T .			Pointer to Driver Pre-Compiler	void	NA
PWM_Stop public Function Reentrant Syncronus Non-Recursion void NA void NA BCM_Handler_Init public Function Non-reentrant Syncronus Non-Recursion void NA void NA BCMH_Get_DR_State public Function Reentrant Syncronus Non-Recursion state pointer to current SW State pointer to current SW State pointer to current SW State Std_Type return Job Status (OK or success, NOK in Failure) BCMH_Get_DR_State public Function Reentrant Syncronus Non-Recursion state pointer to current Door State Std_Type return Job Status (OK or success, NOK in Failure) BCMH_Update_SW_State public Function Non-reentrant Syncronus Non-Recursion void NA Std_Type Status (OK or success, NOK in Failure) BCMH_Update_SM_State public Function Non-reentrant Syncronus Non-Recursion void NA Std_Type Status (OK or success, NOK in Failure) BCMH_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion void NA Std_Type Failure) BCMH_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion void NA Std_Type Failure) BCMH_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion void NA Std_Type Failure) BCMH_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion void NA Std_Type Failure) BCMI_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion void NA Std_Type Failure) BCM_Init public Function Non-reentrant Syncronus Non-Recursion void Na Std_Type Or success, NOK in Failure) Pointer to Module Pre-Compiler or void Na Std_Type Or success, NOK in Failure) Pointer to Module Pre-Compiler or void Na Na Std_Type Or success, NOK in Failure) Pointer to Module Pre-Compiler or void Na Na Std_Type Or success, NOK in Failure) Pointer to Module Pre-Compiler or void Na Na Std_Type Or success, NOK in Failure)			PWM_Start	public	Function	Reentrant	Syncronus	Non-Recursion	channelld		void	NA
BCM_Get_SW_State			PWM_Stop	public	Function	Reentrant	Syncronus		channelld	Identify controlled PWM channel Id	void	NA
BCMH_Get_SW_State public Function Reentrant Syncronus Non-Recursion *state pointer to current SW State Std_Type or success, NOK in Failure) BCMH_Get_DR_State public Function Reentrant Syncronus Non-Recursion *state pointer to current Door State Std_Type or success, NOK in Failure) BCMH_Get_SPD_State public Function Reentrant Syncronus Non-Recursion Non-Recursion *state pointer to current SPD State Public Function Status (OK or success, NOK in Failure) BCMH_Update_SW_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCMH_Update_DR_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCMH_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCMH_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type or success, NOK in Failure) BCMH_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type or success, NOK in Failure) BCMH_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type or success, NOK in Failure) BCM_Init Public Function Non-reentrant Syncronus Non-Recursion *ConfigPtr Configured Pre-Compiler configured data Void NA return Job Status (OK or success, NOK in Failure)	ECU 2		BCM_Handler_Init	public	Function	Non-reentrant	Syncronus	Non-Recursion	void	NA	void	NA
BCM_Handler BCM_Get_DR_State public Function Reentrant Syncronus Non-Recursion *state pointer to current Door State Std_Type or success, NOK in Failure) BCM_Get_SPD_State public Function Reentrant Syncronus Non-Recursion *state pointer to current SPD State Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SW_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_DR_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State Public Function Non-reentrant Syncronus		BCM_Handler	BCMH_Get_SW_State	public	Function	Reentrant	Syncronus	Non-Recursion	*state	pointer to current SW State	Std_Type	
BCM_Handler BCM_Get_SPD_State public Function Reentrant Syncronus Non-Recursion *state pointer to current SPD State Std_Type or success, NOK in Failure) BCM_Update_SW_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_DR_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Update_SPD_State public Function Non-reentrant Syncronus Non-Recursion Void NA Std_Type return Job Status (OK or success, NOK in Failure) BCM_Init public Function Non-reentrant Syncronus Non-Recursion ConfigPtr Pointer to Module Pre-Compiler configured data Void NA NA NA NA NA NA NA N			BCMH_Get_DR_State	public	Function	Reentrant	Syncronus	Non-Recursion	*state	pointer to current Door State	Std_Type	
BCMH_Update_SW_State			BCMH_Get_SPD_State	public	Function	Reentrant	Syncronus	Non-Recursion	*state	pointer to current SPD State	Std_Type	return Job Status (OK in cause or success, NOK in case of Failure)
BCMH_Update_DR_State			BCMH_Update_SW_State	public	Function	Non-reentrant	Syncronus	Non-Recursion	void	NA	Std_Type	
BCMH_Update_SPD_State			BCMH_Update_DR_State	public	Function	Non-reentrant	Syncronus	Non-Recursion	void	NA	Std_Type	
SOM_IIII PUBLIC FUNCTION NOIT-rectuls on Configured data Configured data *msgld Pointer to Message ID to be sent return Job Status (OK			BCMH_Update_SPD_State	public	Function	Non-reentrant	Syncronus	Non-Recursion	void		Std_Type	return Job Status (OK in cause or success, NOK in case of Failure)
* msgld Pointer to Message ID to be sent return Job Status (OK			BCM_Init	public	Function	Non-reentrant	Syncronus	Non-Recursion	* ConfigPtr		void	· ·
- Icam ood cialad (cr						Non-reentrant	Syncronus	Non-Recursion	* msgld			return Job Status (OK in cause
BCM_Send public Function Non-reentrant Syncronus Non-Recursion *Handleld Message to Handelr moduile Id StdType or success, NOK in		всм	BCM_Send pub	public	public Function						StdType	or success, NOK in case of
								*data	•	1 "		
* msgld Pointer to Message ID to be received									* msgld	Pointer to Message ID to be received		raturn Inh Status (OK in cause

Detailed API Description											
ECU name	SWC name	API Description	Scope	Туре	Reenterency	Syncronization	Recursion	Input Parameters	Parameter Description	Return Parameters	Parameter Description return Job Status (UK in cause
		BCM_Receive	public	Function	Reentrant	Syncronus	Non-Recursion	*Handleld	Message to Handelr upper Application moduile Id	StdType	or success, NOK in case of Failure)
								*data	Pointer to Data to be received		r dildre)
		CANTransv_Init	public	Function	Non-reentrant	Syncronus	Non-Recursion	* ConfigPtr	Pointer to Driver Pre-Compiler configured data	void	NA
Common Components	CAN_Transv	CANTransv_Send	public	Function	Non-reentrant	Syncronus	Non-Recursion	* data	Pointer to data to be sent	StdType	return Job Status (OK in cause or success, NOK in case of Failure)
		CANTransv_Receive	public	Function	Reentrant	Syncronus	Non-Recursion	* data	Pointer to Data to be received	StdType	return Job Status (OK in cause or success, NOK in case of Failure)
	PORT	PORT_Init	public	Function	Non-reentrant	Syncronus	Non-Recursion	* ConfigPtr	Pointer to Driver Pre-Compiler configured data	void	NA
		DIO_Init	public	Function	Non-reentrant	Syncronus	Non-Recursion	* ConfigPtr	Pointer to Driver Pre-Compiler configured data	void	NA
	DIO	Dio_ReadChannel	public	Function	Reentrant	Syncronus	Non-Recursion	channelld	Identify controlled DIO channel Id	StdType	return Job Status (OK in cause or success, NOK in case of Failure)
		Dio_WriteChannel	public	Function	Non-reentrant	Syncronus	Non-Recursion	channelld State	Identify controlled DIO channel Id define intended write DIO State	StdType	return Job Status (OK in cause or success, NOK in case of Failure)