Detailed Design (DD)

External Interrupt

Authors: Burners Team (Ahmed Salah, Basem Moufreh, Hassan El Gabass , Hazem El Morshedi , Mohamed Safwat).

Customer: NTI

Instructor: Mahmoud Ali, Ahmed Abd El Reheem.

APIs

1. Initialization API
2. Get the interrupt flag API
3. Interrupt call back API

1 Initializing API

Name	Initialization Function
Prototype	STD EXTI_void_init(EXTI_init_config_struct* EXTI_config);
Parameter in	EXTI_config : Pointer to Configuration struct that contains Enable values of INT0, INT1, INT2,Besides the sense level of each
Parameter out	-
Parameter in-out	-
Return Type	STD E_OK \ STD E_NOT_OK
Description	Function Initiate the External interrupt peripheral using the given configurations.
Covered Requirement	[SRS_EXINT_1]

2 Get the interrupt flag API

Name	Get the interrupt flag Function
Prototype	u8 EXTI_get_INT_flag(u8 copy_INT_id);
Parameter in	copy_INT_id: Interrupt id of the selected interrupt.
Parameter out	-
Parameter in-out	-
Return Type	u8
Description	Function return value of the INT flag in case of Polling not interrupting
Covered Requirement	[SRS_EXINT_2]

3 Interrupt call back API

Name	Interrupt call back
Prototype	STD EXTI_void_INT_callback(u8 copy_INT_id, void (*ptr_callback_function)(void));
Parameter in	copy_INT_id : Interrupt id of the selected interrupt ptr_callback_function : Pointer to the callback function that needs to be executed when the flag is raised.
Parameter out	-
Parameter in-out	-
Return Type	STD E_OK \ STD E_NOT_OK
Description	Function address the call back functions to be called by the ISR of each Interrupt.
Covered Requirement	[SRS_EXINT_3]