Moving Robot APIs

(Heba Ramadan Taha)

1- MCAL Layer

A. DIO APIs:

Function Name	DIO_eSetPinDirection(PinId_t PinIdCpy , PinDir_t PinDirCpy)		
Arguments	Inputs	PinIdCpy	enumeration
		description: Dio pin number to set direction	
		PinDirCpy	enumeration
		description: The direction of pin as input or output	
	Outputs	N/A	
		description:	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to set pin direction as input or output		

Name	PinIdCpy		
Type	enumeration		
Rang	DIO_PIN0	0	For pin 0
	DIO_PIN1	1	For pin 1
	.	.	.
	.	.	.
	.	.	.
	DIO_PIN31	31	For pin 31
Description	These values are to determine which pin in MC to be affected by the function		

Name	PinDirCpy		
Type	enumeration		
Rang	DIO_OUTPUT	1	To be output
	DIO_INPUT	0	To be input
Description	These values are to determine the direction of pin as output or input		

Function Name	DIO_eSetPinValue(PinId_t PinIdCpy , PinVal_t PinValCpy)		
Arguments	Inputs	PinIdCpy	enumeration
		description: The pin number to set value	
		PinValCpy	enumeration
		description: The direction of pin as high or low	
	Outputs	N/A	
		description:	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to set pin value high or low		

Name	PinIdCpy		
Type	enumeration		
Rang	DIO_PIN0 DIO_PIN1 . . . DIO_PIN31	0 1 . . . 31	For pin 0 For pin 1 . . . For pin 31
Description	These values are to determine which pin in MC to be affected by the function		

Name	PinValCpy		
Type	enumeration		
Rang	DIO_HIGH	1	To make pin high
	DIO_LOW	0	To make pin low
Description	These values are to determine the value of pin as high or low		

Function Name	DIO_eGetPinValue(PinId_t PinIdCpy , u8 * pPinVal)		
Arguments	Inputs	PinIdCpy	enumeration
		description: The pin number to get value	
	Outputs	pPinVal	u8 *
		description: pointer to location which save value	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to get pin value high or low		

Name	PinIdCpy		
Type	enumeration		
Rang	DIO_PIN0 DIO_PIN1 . . . DIO_PIN31	0 1 . . . 31	For pin 0 For pin 1 . . . For pin 31
Description	These values are to determine which pin in MC to be affected by the function		

B. Timer APIs:

Function Name	TIMER_eInit(ChannelId_t ChIdCpy)		
Arguments	Inputs	ChIdCpy	enumeration
	description: The Channel ID to initialize as Timer		
	Outputs	N/A	
	description:		
	Input/Output	N/A	
	description:		
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to Initialize Timer as specified in the configuration file. Selecting timer hardware based on ChannelID.		

Name	ChIdCpy		
Type	enumeration		
Rang	TIMER_CHANNEL_0	0	For Timer0
	TIMER_CHANNEL_1	1	For Timer1
	TIMER_CHANNEL_2	2	For Timer2
Description	To determine which timer to be affected by the function		

Function Name	TIMER_eStart(ChannelId_t ChIdCpy , u16 TimeCountCpy)		
Arguments	Inputs	ChIdCpy	enumeration
		description: The Channel ID to Start Timer	
		TimeCountCpy	u16
		description: Number of counts in ms	
	Outputs	N/A	
		description:	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to Start timer and count from 0 till number of counts in msec		

Name	ChIdCpy		
Type	enumeration		
Rang	TIMER_CHANNEL_0	0	For Timer0
	TIMER_CHANNEL_1	1	For Timer1
	TIMER_CHANNEL_2	2	For Timer2
Description	To determine which timer to be affected by the function		

Name	TimeCountCpy	
Type	u16	
Rang	1	Min number 1msec
	60000	Max 1 hour
Description	To determine number of mSec to count by timer	

Function Name	TIMER_eStop(ChannelId_t ChIdCpy)		
Arguments	Inputs	ChIdCpy	enumeration
		description: The Channel ID of Timer	
	Outputs	N/A	
		description:	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to stop Timer		

Name	ChIdCpy		
Type	enumeration		
Rang	TIMER_CHANNEL_0	0	For Timer0
	TIMER_CHANNEL_1	1	For Timer1
	TIMER_CHANNEL_2	2	For Timer2
Description	To determine which timer to be affected by the function		

Function Name	TIMER_eGetStatus(ChannelId_t ChIdCpy , TimStat_t* pTimerState)		
Arguments	Inputs	ChIdCpy	enumeration
		description: The Channel ID to initialize as Timer	
	Outputs	pTimerState	TimStat_t*
		description: return state of timer	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to get current state of timer		

Name	ChIdCpy		
Type	enumeration		
Rang	TIMER_CHANNEL_0	0	For Timer0
	TIMER_CHANNEL_1	1	For Timer1
	TIMER_CHANNEL_2	2	For Timer2
Description	To determine which timer to be affected by the function		

Name	pTimerState		
Type	enumeration		
Rang	TIMER_RUNNIG	0	
	TIMER_STOP	1	
	TIMER_EXPIRED	2	
Description	To describe current state of timer		

C. PWM APIs:

Function Name	PWM_eInit(ChannelId_t ChIdCpy)		
Arguments	Inputs	ChIdCpy	enumeration
		description: The Channel ID of timer to initialize as PWM	
	Outputs	N/A	
		description:	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to Initialize PWM module		

Name	ChIdCpy		
Type	enumeration		
Rang	TIMER_CHANNEL_0	0	For Timer0
	TIMER_CHANNEL_1	1	For Timer1
	TIMER_CHANNEL_2	2	For Timer2
Description	To determine which timer to be affected by the function		

Function Name	PWM_eStart (ChannelId_t ChIdCpy ,u8 DutyCycleCpy , u32 FreqCpy)			
Arguments	Inputs	ChIdCpy		enumeration
		description: The Channel ID to start PWM		
		DutyCycleCpy		u8
		description: the duty cycle of the signal		
		FreqCpy		u32
		description: the frequency of the signal		
	Outputs	N/A		
		description:		
	Input/Output	N/A		
		description:		
Return	E_OK	0		
	E_NOK	1		
Description	Call this API to Start PWM module			

Name	DutyCycleCpy	
Type	u8	
Rang	0	Min value of duty
	100	Max value of duty
Description	Duty cycle of pwm signal	

Name	FreqCpy	
Type	u32	
Rang	0	Min value of Freq
	100000	Max(10usec period)
Description	Frequency of pwm signal	

Name	ChIdCpy		
Type	enumeration		
Range	TIMER_CHANNEL_0	0	For Timer0
	TIMER_CHANNEL_1	1	For Timer1
	TIMER_CHANNEL_2	2	For Timer2
Description	To determine which timer to be affected by the function		

Function Name	PWM_eStop(ChannelId_t ChIdCpy)			
Arguments	Inputs	ChIdCpy	enumeration	
	description: The Channel ID of timer			
	Outputs	N/A		
	description:			
	Input/Output	N/A		
	description:			
Return	E_OK	0		
	E_NOK	1		
Description	Call this API to Stop PWM module			

Name	ChIdCpy		
Type	enumeration		
Rang	TIMER_CHANNEL_0	0	For Timer0
	TIMER_CHANNEL_1	1	For Timer1
	TIMER_CHANNEL_2	2	For Timer2
Description	To determine which timer to be affected by the function		

2- On-Board Layer

A. LCD:

Function Name	LCD_eInit(void)		
Arguments	Inputs	N/A	
	description:		
	Outputs	N/A	
	description:		
	Input/Output	N/A	
	description:		
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to initialize LCD module		

Function Name	LCD_eSendCommand(Cmd_t u8cmdcpy)		
Arguments	Inputs	u8cmdcpy	enumeration
	description: a copy of the command to send to the lcd		
	Outputs	N/A	
	description:		
	Input/Output	N/A	
	description:		
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to send command to set LCD		

Name	u8cmdcpy	
Type	enumeration	
Rang	LCD_8BIT	0
	LCD_DISPLAY_ON	1
	LCD_DISPLAY_OFF	2
	LCD_CLEAR	3
	LCD_ENTRY_MODE	4
Description	These values are the commands to be sent to lcd .	

Function Name	LCD_eSendChar(u8 u8charcpy)		
Arguments	Inputs	u8charcpy	u8
		description: a copy of the Data to send on the lcd	
	Outputs	N/A	
		description:	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to send Data on LCD		

Name	u8charcpy
Type	u8
Rang	{0 ,….,127}
Description	The decimal representation of ASCII code.

Function Name	LCD_eSetPosition(u8 u8ColCpy , u8 u8RowCpy)		
Arguments	Inputs	u8ColCpy	u8
		description: the horizontal position starting from 0:15 for 2x16 lcd	
		u8RowCpy	u8
		description: the vertical position (0:1) for 2x16 lcd	
	Outputs	N/A	
		description:	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to go to specific position on the lcd		

Name	u8ColCpy	
Type	u8	
Rang	0	The first position in the screen starting from left
	15	The last position in the screen starting from left
Description	These values are the horizontal positions in a 2x16 LCD.	

Name	u8RowCpy	
Type	u8	
Rang	0	The first row in the screen starting from upper row
	1	The second row in the screen starting from upper row
Description	These values are the vertical positions in a 2x16 LCD	

B. MOTOR:

Function Name	MOTOR_eInit(PinId_t MotorIdCpy , PinId_t SpeedPincpy, ChId_t PWM_Ch)		
Arguments	Inputs	MotorIdCpy	enumeration
		description: The motor existence dio pin	
		SpeedPincpy	enumeration
		description: The motor speed(PWM) pin	
		PWM_Ch	enumeration
		description: The PWM channel dio pin	
	Outputs	N/A	
		description:	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to initialize Motor module (initialize dio pin as output and pwm_Init() with specific channel of timer)		

Name	MotorIdCpy		
Type	enumeration		
Rang	DIO_PIN0	0	For pin 0
	DIO_PIN1	1	For pin 1
	.	.	.
	.	.	.
	.	.	.
	DIO_PIN31	31	For pin 31
Description	These values are to determine which pin in MC to be affected by the function		

Function Name	MOTOR_eStartt(PinId_t MotorIdCpy , u8 MotorSpeed, ChId_t PWM_Ch)		
Arguments	Inputs	MotorIdCpy	enumeration
		description: The motor existence dio pin	
		MotorSpeed	u8
		description: The motors peed dio pin	
		PWM_Ch	enumeration
		description: The PWM channel dio pin	
	Outputs	N/A	
		description:	
	Input/Output	N/A	
		description:	
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to start Motor with specific speed		

Name	MotorIdCpy		
Type	enumeration		
Rang	DIO_PIN0	0	For pin 0
	DIO_PIN1	1	For pin 1
	.	.	.
	.	.	.
	DIO_PIN31	31	For pin 31
Description	Specific dio pin for motor		

Name	MotorSpeed	
Type	u8	
Rang	0	Min speed
	80	Max speed for robot
Description	Speed of motor	

Function Name	MOTOR_eStop(PinId_t MotorIdCpy)			
Arguments	Inputs		MotorIdCpy	enumeration
			description: The motor existence dio pin	
	Outputs		N/A	
			description:	
	Input/Output		N/A	
			description:	
Return	E_OK		0	
	E_NOK		1	
Description	Call this API to stop Motor module			

Name	MotorIdCpy		
Type	enumeration		
Rang	DIO_PIN0 DIO_PIN1 . . . DIO_PIN31	0 1 . . . 31	For pin 0 For pin 1 . . . For pin 31
Description	These values are to determine which pin in MC to be affected by the function		

3- APP Layer

A. Robot Module:

Function Name	Robot_eInit(void)		
Arguments	Inputs	N/A	
	description:		
	Outputs	N/A	
	description:		
	Input/Output	N/A	
	description:		
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to initialize the Robot Control module and the needed other modules		

Function Name	Robot_eUpdateMoving(void)		
Arguments	Inputs	N/A	
	description:		
	Outputs	N/A	
	description:		
	Input/Output	N/A	
	description:		
Return	E_OK	0	
	E_NOK	1	
Description	Call this API to periodically to update moving direction and speed of motor.		