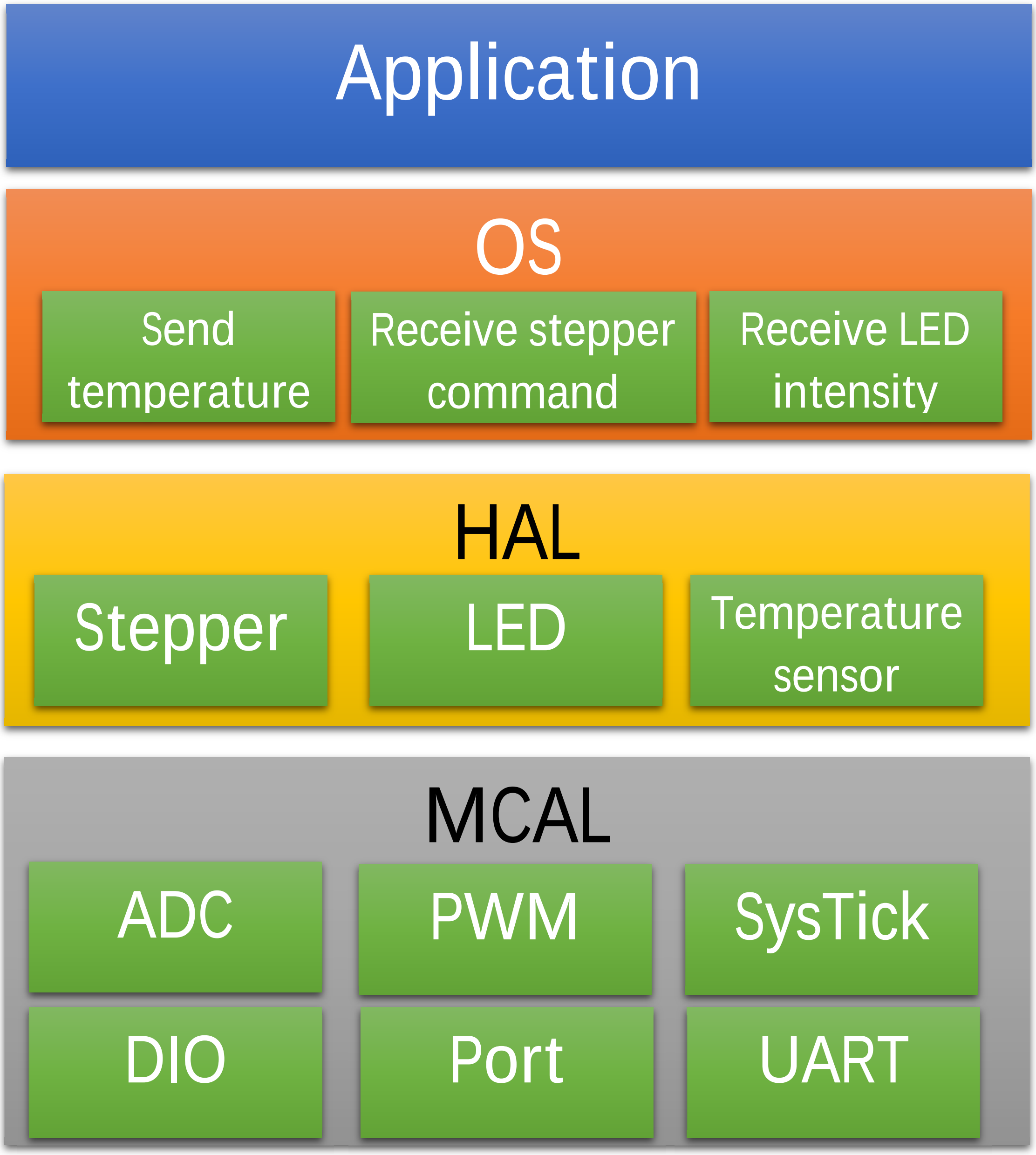
Group 9

Software layers report

Tiva 1 (Software layers)



Application

OS

Send temperature

Receive stepper command

Receive LED intensity

HAL

Stepper

LED

Temperature sensor

MCAL

ADC

PWM

SysTick

DIO

Port

UART

1. MCAL Layer

a) Port driver

Function definition:

Name	Port_Init	
Input	uint8	port_index
Return	void	
Description	Initialize port based on selected port_index (0 to 5) by enabling the clock, unlocking the port, and making the selected mode digital	

Name	Port_SetPinDirection	
Input	uint8	port_index
	uint8	pins_mask
	Port_PinDirectionType	pins_direction
Return	void	
Description	Change the direction of the selected pins by pins_mask in the port selected by port_index	

b) DIO driver

Function definition:

Name	DIO_ReadPort	
Input	uint8	port_index
	uint8	pins_mask
Return	uint8	pins_level
Description	Return the value of the pins selected by pins_masks in the port selected by port_index	

Name	DIO WritePort	
Input	uint8	port_index
	uint8	pins_mask
	uint8	pins_level
Return	void	
Description	Change the value of the pins selected by pins_masks in the port selected by port_index to input pins_level.	

c) ADC driver

Function definition:

Name	ADC0_Init
Input	void
Return	void
Description	Enable clock to ADC, give priority to sequencer 3, select processor as trigger source, make ADC read from inner temperature sensor, enable analog mode and alternative function select.

Name	ADC0_SS3_In	
Input	Uint16*	data
Return	void	
Description	Read converted value from ADC FIFO and return it to data.	

d) PWM driver

Function definition:

Name	PWM1_Init
Input	void
Return	void
Description	Enable clock to pwm, use pwm divisor by 64 Enable PWM to output pin, set initial value to load and compare registers, take action when counter is reloaded and when counter reaches compA while counting down.

Name	SetDutyCycle	
Input	uint8	Duty_cycle
Return	void	
Range	0 up to 100 percent	
Description	Overwrites compare A register to modify duty cycle.	

e) SysTick driver

Function definition:

Name	Systick_init
Input	void
Return	void
Description	Put max value in Reload register, put zero in Current register, in control register enable systick and use internal clock source.

Name	Systick_wait1ms	
Input	void	
Return	void	
Description	Put in reload register number of clock cycles per one ms (80000 in case of PLL), wait until current register reaches to zero and count bit in ctrl register is set to 1.	

f) UART drivers

Function definition:

Name	UART7_init	
Input	void	
Return	void	
Description	Enable clock to UART, use clock divisor 16, use baud rate 9600, enable FIFO, use word length of 8 bit, enable alternative function select, enable PCTL for UART	

Name	UART7_available	
Input	void	
Return	uint8	UART_availability
Description	Check if the FIFO of receiver is empty to receive data	

Name	UART7_sendbyte	
Input	uint8	data
Return	void	
Description	Check if the FIFO of transmitter isn’t full, then write data to UART data register.	

Name	UART7_receivebyte	
Input	void	
Return	uint8	data
Description	Wait until UART is available to receive , then return UART data register.	

2. HAL Layer

a) Temperature sensor driver

Function definition:

Name	Temperature_Sensor_Init
Input	void
Return	void
Description	Enable the TS bit included in called ADC init function to make the adc read value from temoerature sensor not from the input channels.

Name	Read_Temp_Sensor_Value	
Input	void	
Return	uint8_t	Temperature value
Description	Calculate the temperature value in Celsius by insert the value from data argument in ADC0_SS3_In into temperature equation .	

b) Stepper motor driver

Function definition:

Name	stepper_init
Input	void
Return	void
Description	Enable port d to have 4 pins of motor driver Included in port init function for port d And initialize the first half step avoiding error in .

Name	step_forword
Input	void
Return	void
Description	Make the stepper rotate completely 30 degrees clockwise

Name	step_backword
Input	void
Return	void
Description	Make the stepper rotate completely 30 degrees anti clockwise

c)LED driver

Function definition:

Name	LED_Init
Input	void
Return	void
Description	Enable PF4 PIN in Portf to be alternative function as analog pin And enable the PWM Device by Calling PWM init function .

Name	LED_Intensity	
Input	uint8	duty_cycle
range	0 up to 100 for intensity percentage	
Return	void	
Description	Determine the intensity of the led .	

3. OS Layer

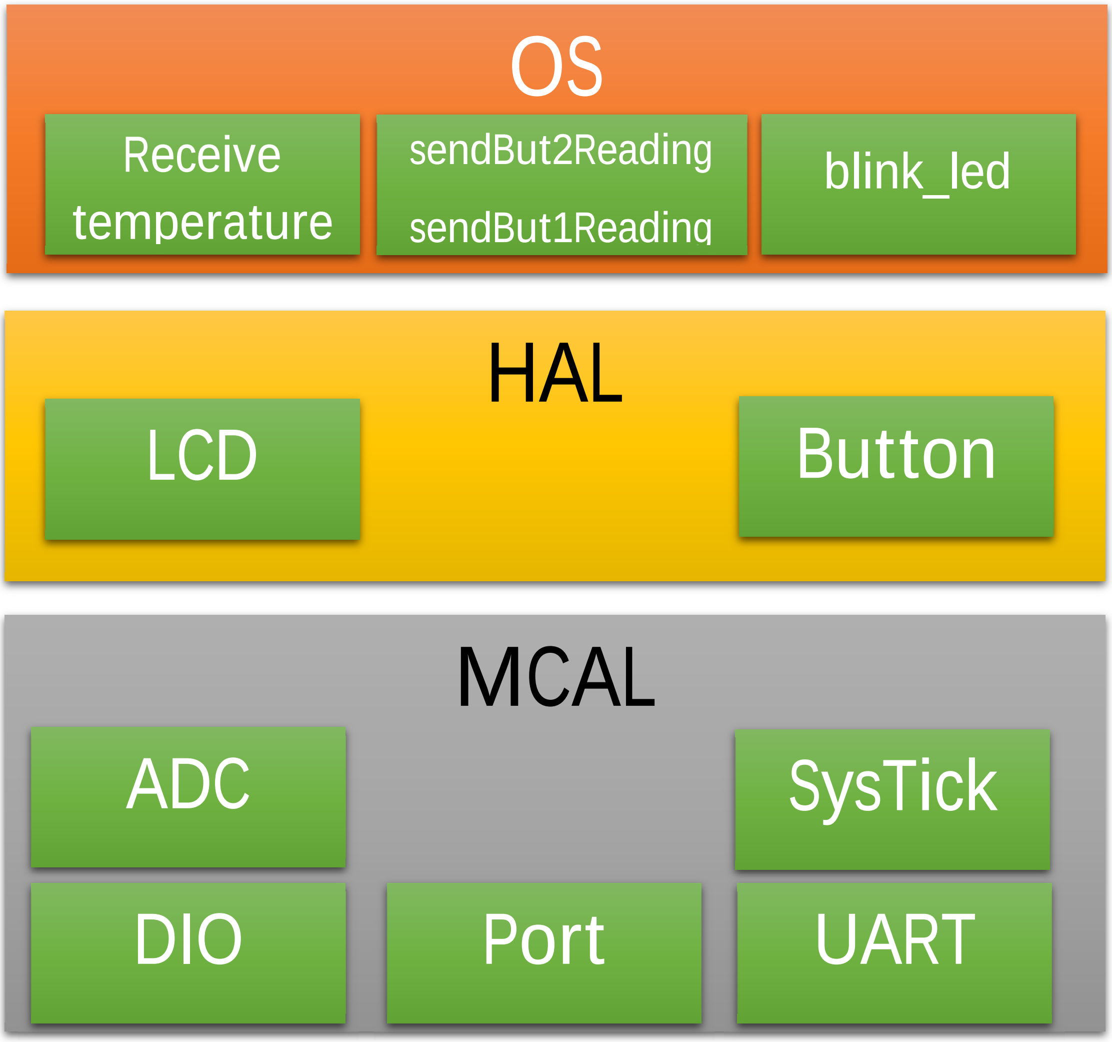
Function definitions:

Name	Send_temperature
Input	void* pointer
Return	void
Description	Read temperature from sensor from (read_temp_sensor_value) and send it via UART7

Name	Receive_stepper_command
Input	void* pointer
Return	void
Description	If we receive '*' from UART7 then the coming data is stepper command, if we receive '1' then rotate 30° clockwise, and if we receive '3' then rotate 30° anticlockwise

Name	Receive_LED_intensity
Input	void* pointer
Return	void
Description	If we receive '$' from UART7 then the coming data is LED intensity, then we write intensity in 'LED_intensity' function.

Tiva 2 (Software layers)



1.MCAL Layer

a) Port driver

Function definition:

Name	Port_Init	
Input	uint8	port_index
Return	void	
Description	Initialize port based on selected port_index (0 to 5) by enabling the clock, unlocking the port, and making the selected mode digital	

Name	Port_SetPinDirection	
Input	uint8	port_index
	uint8	pins_mask
	Port_PinDirectionType	pins_direction
Return	void	
Description	Change the direction of the selected pins by pins_mask in the port selected by port_index	

b) DIO driver

Function definition:

Name	DIO_ReadPort	
Input	uint8	port_index
	uint8	pins_mask
Return	uint8	pins_level
Description	Return the value of the pins selected by pins_masks in the port selected by port_index	

Name	DIO WritePort	
Input	uint8	port_index
	uint8	pins_mask
	uint8	pins_level
Return	void	
Description	Change the value of the pins selected by pins_masks in the port selected by port_index to input pins_level.	

C) ADC driver

Function definition:

Name	ADC0_Init
Input	void
Return	void
Description	Enable clock to ADC, give priority to sequencer 3, select processor as trigger source, make ADC read from inner temperature sensor, enable analog mode and alternative function select.

Name	ADC0_SS3_In	
Input	Uint16_t	data
Return	void	
Description	Read converted value from ADC FIFO and return it to data.	

D) SysTick driver

Function definition:

Name	Systick_init
Input	void
Return	void
Description	Put max value in Reload register, put zero in Current register, in control register enable systick and use internal clock source.

Name	Systick_wait1ms
Input	void
Return	void
Description	Put in reload register number of clock cycles per one ms (80000 in case of PLL), wait until current register reaches to zero and count bit in ctrl register is set to 1.

d) UART drivers

Function definition:

Name	UART7_init
Input	void
Return	void
Description	Enable clock to UART, use clock divisor 16, use baud rate 9600, enable FIFO, use word length of 8 bit, enable alternative function select, enable PCTL for UART

Name	UART7_available	
Input	void	
Return	uint8	UART_availability
Description	Check if the FIFO of receiver is empty to receive data	

Name	UART7_sendbyte	
Input	uint8	data
Return	void	
Description	Check if the FIFO of transmitter isn’t full, then write data to UART data register.	

Name	UART7_receivebyte	
Input	void	
Return	uint8	data
Description	Wait until UART is available to receive , then return UART data register.	

2. HAL Layer

a) Button driver

Function definition:

Name	isPressed
Input	Uint8 Port_index
	Uint8 Pin_Number
Return	Uint8 value

b) LCD driver

Function definition:

Name	LCD_Init
Input	void
Return	void
Description	Initialize LCD as 2*16

Name	LCD_sendcommand	
Input	uint8	command
Return	void	
Description	Send commands to the LCD	

Name	LCD_displayStringRowColumn	
Input	uint8	row
	uint8	col
	const char *	str
Return	void	
Description	Display string at a certain column and row	

Name	LCD_clearscreen
Input	void
Return	void
Description	Clear LCD screen

3. OS Layer

Function definitions:

Name	sendBut1Reading
Input	void* pointer
Return	void
Description	Read Button 1 from Tiva C and send it via UART7

Name	sendBut2Reading
Input	void* pointer
Return	void
Description	Read Button 2 from Tiva C and send it via UART7

Name	sendPotReading
Input	void* pointer
Return	void
Description	Read Potentiometer reading and convert it using ADC into digital values and send it via UART7

Name	ReceiveTemp
Input	void* pointer
Return	void
Description	If we receive '*' from UART7 then the coming data is Tempreture value, then we write Value on LCD