RDL Data Logger API Manual

1. DL_Config_Digital_Outputs

Function Prototype	void DL_Config_Digital_Outputs(void)
Description	Configures digital pins D46, D47, D48 and D49 as OUTPUT
Parameters	None
Return Type	None
Function call Example	DL_Config_Digital_Outputs( );

2. DL_Digital_Inputs

FunctionPrototype	void DL_Digital_Inputs(void)
Description	Configures digital pins D37, D38, D40, D41, D42, D43, D44, D45 as INPUT
Parameters	None
Return Type	None
Function call Example	DL_Digital_Inputs( );

3. DL_Analog_Read

Function Prototype	int DL_Analog_Read(uint8_t pin)
Description	Reads analog pins A0,A1,A2
Parameters	pin - One of the pin number mentioned above
Return Type	ADC value
Function call Example	Int adcValue = DL_Analog_Read(A0);

4. DL_Analog_Write

Function Prototype	void DL_Analog_Write(uint8_t pin, uint8_t value)
Description	Write analog pins A0,A1,A2
Parameters	pin - One of the pin number mentioned above
Return Type	Value - Value to be written(0-255)
Function call Example	DL_Analog_Write(A0, 255);

5. DL_Config_Digital_IO

FunctionPrototype	void DL_Config_Digital_IO(uint8_t pin, uint8_t direction)
Description	Configures the port (Applicable to D37, D38, D40, D41, D42, D43, D44 and D45)
Parameters	pin[IN] - One of the pin number mentioned above direction[IN] - OUTPUT/INPUT
Return Type	None

Function call Example	DL_Config_Digital_IO(D37, INPUT);
	DL_Config_Digital_IO(D37, OUTPUT);

6. DL_Digital_IO_Write

Function Prototype	void DL_Digital_IO_Write(uint8_t pin, uint8_t value)
Description	Writes the IO pins D46, D47, D48 and D49 as OUTPUT
Parameters	pin[IN] -One of the pin number mentioned above value[IN] - High/low
Return Type	None

Function call Example	DL_Digital_IO_Write(D46, HIGH); //D46 pin is high
	DL_Digital_IO_Write(D46, LOW);

7. DL_Digital_IO_Read

Function Prototype	uint8_t DL_Digital_IO_Read(uint8_t pin)
Description	Reads the IO pin (Applicable to D37, D38, D40, D41, D42, D43, D44 and D45)
Parameters	pin[IN] - One of the pin number mentioned above
Return Type	value - High/low
Function call Example	uint8_t Value = DL_Digital_IO_Read(D37);

8. sendATcommand

Function Prototype	static uint8_t sendATcommand(char* ATcommand, char* expected_answer, unsigned int timeout)
Description	Function to send AT Command to GPRS
Parameters	ATcommand(IN) - AT command to be send expected_answer[IN] - Expected response (normally "OK") timeout[IN] - How long to wait for the response
Return Type	Status
Function call Example	uint8_t status = sendATcommand(“AT/r/n” , ”OK/r/n”, 2000);

9. DL_GSM_Send_SMS

Function Prototype	void DL_GSM_Send_SMS(char *numb, char *msg)
Description	Send sms to GSM device
Parameters	numb[IN] - Phone number msg{IN] - Message to be sent
Return Type	None
Function call Example	char num[] = “99xxxxxxxx”; char msg[] = “Hello World”; DL_GSM_Send_SMS(num, msg);

10. DL_GSM_Receive

Function Prototype	void DL_GSM_Receive(char index)
Description	Receive sms to GSM device
Parameters	index[IN] - Memory location number to read the message
Return Type	None
Function call Example	DL_GSM_Receive(2);

11. DL_GSM_MakeCall

Function Prototype	void DL_GSM_MakeCall(char *numb)
Description	Make a Call
Parameters	numb[IN] - Phone number
Return Type	None
Function call Example	char num[] = “99xxxxxxxx” DL_GSM_MakeCall(num);

12. DL_Config_GPRS

Function Prototype	void DL_Config_GPRS(char* APN)
Description	Function to configure GPRS
Parameters	APN[IN] - APN of GSMnetwork operation
Return Type	None
Function call Example	DL_Config_GPRS(“tp://airtelgprs.com”); NOTE: APN is shown for airtel sim,change the APN if other network operator is used.

13. DL_GPRS_Upload_FTP

Function Prototype	void DL_GPRS_Upload_FTP(char* host, char* userName, char* password, char* fileName, char* folder)
Description	Function to Upload files on FTP server
Parameters	host[IN] - Host url userName[IN] - Username password[IN] - Password filename[IN] - FileName to upload the file folder[IN] - Folder name in local pc to fetch the file from
Return Type	None
Function call Example	char host[]=”ftp://kanwal.varmatrix.com”; char user[]=”kanwalabc123”; char pass[]=”xxxxxxxx”; char file[]=”12_7_2018_adc.csv”; char folder[]=”test”; DL_GPRS_Upload_FTP(host, user, pass, file, folder);

14. DL_RTC_Get_Time

Function Prototype	DateTime DL_RTC_Get_Time(void)
Description	This function Gets RTC date and Time
Parameters	None
Return Type	Datetime – Structure holding the calendar

Function call Example	DateTime calendar;
	calendar =DL_RTC_Get_Time();

15. DL_GPRS_Upload_JSON

Function Prototype	void DL_GPRS_Upload_JSON(char* data, char* url, char* ext, char* type)
Description	Function to Upload log data using JSON method
Parameters	data[IN] - Data to be pushed to server url[IN] - Url of the server ext[IN] - Url extension type[IN] - Application/json or application/x-www-form-urlencoded
Return Type	None
Function call Example	char data[]=”\”id\”:\”Hello world\” “; char url[]=”iotpi.in”; char ext[]=”/rfidwebserver/rfidinsert.php”); char type[]=”application/json”;

	DL_GPRS_Upload_JSON(data, url, ext, type);

16.DL_GPRS_Upload_MQTT

Function Prototype	void DL_GPRS_Upload_MQTT(char* data, char* host, char* port, char* username, char* password, char* topic)
Description	Function to Upload log data using MQTT protocol
Parameters	data[IN] - Data to be pushed to cloud host[IN] - MQTThost port[IN] - MQTTport number username[IN] - MQTTusername password[IN] - MQTTpassword topic[IN] - MQTTtopic
Return Type	None
Function call Example	char data[] =”Hello world”; char host[] =”m12.cloudmqtt.com”; char port[] =”11068”; char user[]=”iihbfshs”; char pass[]=”xxxxxxxxxxx”; char topic[]=”kanwal”; DL_GPRS_Upload_MQTT(data, host, port, user, pass, topic);

17.DL_Modbus_Master_Read_Raw

Function Prototype	void DL_Modbus_Master_Read_Raw(uint8_t funCode, uint8_t slave_id, uint16_t start_addr, uint16_t len, uint8_t *data)
Description	This function sends data using modbus protocol
Parameters	funCode [IN] - Function code (readCoils/ readDiscreteInputs/ readInputRegisters/ readHoldingRegisters) slave_id [IN] - Slave id of the device start_addr [IN] - Starting address to read data frame len [IN] - Length of the data data [OUT] - Read data
Return Type	None

Function call Example	uint8_t data[50]={0};
	DL_Modbus_Master_Read_Raw(2, 8, 40001, 5,data);

18. DL_SD_Write

Function Prototype	void DL_SD_Write(char *fileName, char *data)
Description	This function write data to SD card file
Parameters	filename[IN] - Filename to write to data{IN] -Data to be written in file
Return Type	None
Function call Example	DL_SD_Write("test_RDL_DL.txt", "Hello World\n"); NOTE: max 255 bytes can be written once

19. DL_SD_Read

Function Prototype	void DL_SD_Read(char *fileName, uint8_t *data)
Description	This function Reads data from SD card file
Parameters	filename[IN] - Filename to write to data{Out} -Data to be written in file
Return Type	None

Function call Example	uint8_t data[50]={0};
	DL_SD_Read("test_RDL_DL.txt", data);
	NOTE: max 255 bytes can be read once

20. DL_RTC_Set_Time

Function Prototype	void DL_RTC_Set_Time(uint16_t year, uint8_t month, uint8_t date, uint8_t hour, uint8_t min, uint8_t sec)
Description	This function Sets RTC date and Time
Parameters	Year[IN] - Year Month [IN] - Month of the year Date[IN] - Date of the month Hour[IN] - Hour Min[IN] - Minute Sec[IN] - Second
Return Type	None
Function call Example	DL_RTC_Set_Time(2018, 11, 16, 11, 57, 2);

21. DL_FRAM_Read

Function Prototype	void DL_FRAM_Read (uint16_t framAddr, uint8_t* data, int len)
Description	Function to read data from FRAM
Parameters	framAddr[IN] - 16 bit address to read the data from data[OUT] - Pointer to the data read len[IN] - Number of bytes to read
Return Type	None

Function call Example	uint8_t data[50] ={0};
	DL_FRAM_Read (0x0010, data, 10);

22. DL_FRAM_Write

Function Prototype	void DL_FRAM_Write(uint8_t* data, uint16_t framAddr)
Description	Function to store data in FRAM
Parameters	data[IN] -Pointer to the data to be written framAddr[IN] - 16 bit address to write the data to
Return Type	None
Function call Example	uint8_t data =”Hello world”; DL_FRAM_Write(data, 0x0010);

23. DL_PWM

Function Prototype	void DL_PWM(uint8_t pin,uint8_t ms)
Description	Function for PWM - fading
Parameters	pin[IN] -configures D46; ms[IN] - Delay in ms
Return Type	None
Function call Example	DL_PWM(D46,30);

24. DL_SRAM_Init

Function Prototype	bool DL_SRAM_Init(void)
Description	Function to Init Flash
Parameters	None
Return Type	Status

Function call Example	bool status =0;
	status =DL_SRAM_Init();

25. DL_SRAM_Read_JDEC_Id

Function Prototype	void DL_SRAM_Read_JDEC_Id(uint8_t *buf)
Description	Function to read JDEC id
Parameters	Buf[IN] - Pointer to hold JDEC ID
Return Type	None

Function call Example	uint8_t JDEC_Id[10]={0};
	DL_SRAM_Read_JDEC_Id(JDEC_Id);

26. DL_SRAM_Write

Function Prototype	void DL_SRAM_Write(uint32_t addr, uint8_t *buf, uint32_t data_length)
Description	Function to write into Flash
Parameters	Buf[IN] - Pointer to data to be written data_length[IN] - Length of data to be written
Return Type	None
Function call Example	uint8_t buff[]=”Hello world”; DL_SRAM_Write(0x00000010,buff,strlen(buff));

27. DL_SRAM_Wait

Function Prototype	int DL_SRAM_Wait(void)
Description	Function to wait for data write/erase to complete
Parameters	None
Return Type	Status

Function call Example	Int status =0;
	Status =DL_SRAM_Wait();

28. DL_SRAM_Read

Function Prototype	void DL_SRAM_Read(uint32_t addr, uint8_t *buf , uint32_t len)
Description	Function to read data
Parameters	Addr[IN] - Address to read data from len[IN] - Length of data to be read Buf[OUT] - Pointer to data to be read
Return Type	None

Function call Example	uint8_t buff[50] ={0};
	DL_SRAM_Read(0x00000010, buff, 10);

29. DL_SRAM_Chip_Erase

Function Prototype	void DL_SRAM_Chip_Erase(void)
Description	Function to erase the entire chip
Parameters	None
Return Type	None
Function call Example	DL_SRAM_Chip_Erase();

30. DL_SRAM_BlockErase4K

Function Prototype	void DL_SRAM_BlockErase4K(uint32_t addr)
Description	Function to erase data in 4k blocks(sector)
Parameters	Addr[IN] - Address to erase the sector
Return Type	None
Function call Example	DL_SRAM_BlockErase4K(0x00000000);

31. DL_Serial_Read_Byte

Function Prototype	char DL_Serial_Read_Byte(void)
Description	Function to erase data in 4k blocks(sector)
Parameters	None
Return Type	Byte of data

Function call Example	char inByte=0;
	inByte =DL_Serial_Read_Byte();

32. DL_Serial_Write_Byte

Function Prototype	void DL_Serial_Write_Byte(uint8_t byte)
Description	Function to read and write serial data
Parameters	Byte[IN] - 1 Byte to write
Return Type	None
Function call Example	DL_Serial_Write_Byte(‘H’);

33. DL_SPI_Read

Function Prototype	uint8_t DL_SPI_Read(uint32_t clock, uint8_t bitOrder, uint8_t dataMode, uint8_t slavePin)
Description	Function to Read SPI data
Parameters	Clock[IN]- Max clock frequency(speed) of spi BitOrder[IN- MSBFIRST/LSBFIRST dataMode[IN] - SPI_MODE0/SPI_MODE1/SPI_MODE2/ SPI_MODE3 SlavePin[IN] - Chip select/ Slave select pin
Return Type	Read data
Function call Example	DL_SPI_Read(14000000,MSBFIRST,SPI_MODE0,53);

34. DL_I2C_Read

Function Prototype	uint8_t DL_I2C_Read(uint8_t slave_id, uint8_t len, uint8_t* data)
Description	Function to Read I2C data
Parameters	slave_id[IN] - Slave device id len[IN] - Number of bytes to read data[OUT] - Pointer to data read
Return Type	Read data

Function call Example	uint8_t data[10] ={0};
	DL_I2C_Read(12, 5, data);

35. DL_SPI_Write

Function Prototype	void DL_SPI_Write(uint32_t clock, uint8_t bitOrder, uint8_t dataMode, uint8_t data, uint8_t slavePin)
Description	Function to Write data on SPI bus
Parameters	clock[IN] - Max clock frequency(speed) of spi bitOrder[IN] - MSBFIRST/LSBFIRST dataMode[IN]- SPI_MODE0/SPI_MODE1/SPI_MODE2/ SPI_MODE3 data[IN] - Data to be written slavePin[IN] - Chip select/Slave select pin
Return Type	None
Function call Example	DL_SPI_Write(14000000,MSBFIRST,SPI_MODE0,’H’, 53);

36. DL_I2C_Write

Function Prototype	void DL_I2C_Write(uint8_t slave_id, uint8_t len, uint8_t* data)
Description	Function to Write data on I2C bus
Parameters	slave_id [IN] - Slave device id len[IN] - Number of bytes to be written data[IN] - Pointer to Data to be written
Return Type	None
Function call Example	Uint8_t data[]=”Hello world”; DL_I2C_Write(12, strlen(data), data);

37. DL_Reset

Function Prototype	void DL_Reset(uint8_t resetPin)
Description	Function for Soft Reset on Arduino
Parameters	ResetPin[IN] - Pin number which is 38
Return Type	None
Function call Example	DL_Reset(38);

38. DL_PWDT

Function Prototype	void DL_PWDT(uint32_t timeout)
Description	Function for Physical Watch Dog Timer
Parameters	timeout[IN] - Timeout in ms
Return Type	None
Function call Example	DL_PWDT(8000); NOTE: minimum 0,maximum 2mins

39. DL_EEPROM_Write

Function Prototype	void DL_EEPROM_Write(uint8_t addr-, uint8_t val)
Description	Function for writing data into EEPROM
Parameters	Addr[IN] - Address to write data Val[IN] - Value or data to write
Return Type	None
Function call Example	DL_EEPROM_Write(100, 20);

40. DL_EEPROM_Read

Function Prototype	uint8_t DL_EEPROM_Read(uint8_t addr)
Description	Function for reading data from EEPROM
Parameters	Addr[IN] - Address to write data val[IN] - Value or data to write
Return Type	Byte of data
Function call Example	DL_EEPROM_Read(100);