e-Yantra Robotics Competition (eYRC-2018)

Task 1 – Predefined Functions

This document covers the list of functions and it's description available for you to use in your task. These predefined functions are available in NS_Task_1_Predef.h file.

- `forward();`

This function is used to make both motors move in forward direction.

- `back();`

This function is used to make both motors move in backward direction.

- `left();`

This function is used to make the robot move left; where left motor moves in backward direction and right motor moves in forward direction.

- `right();`

This function is used to make the robot move right; where left motor moves in forward direction and right motor moves in backward direction.

- `soft_left();`

This function is used to make the robot move left; where left motor is stationary and right motor moves in forward direction.

- `soft_right();`

This function is used to make the robot move right; where left motor moves in forward direction and right motor is stationary.

- `stop();`

This function is used to stop the robot; where left motor and right motor are stationary.

- `velocity(left_motor, right_motor);`

This functions takes values from 0 to 255 for each motor. The values correspond to the PWM duty cycle.

- `unsigned char ADC_Conversion(unsigned char channel_no);`

This function is used to read values from the white line sensors and front IR proximity sensor. The channel numbers are as follows: 0, 1 and 2 for the left, middle and right sensors respectively, and channel number 4 is for the front IR proximity sensor.

Note: The accurate range of the line sensor is below 100mm and ADC_Conversion, above 100 the values are not to be trusted. When nothing is detected by the Proximity Sensor it returns values in the range 101-140.

❖ `_delay_ms(unsigned int milliseconds);`

This function is used to run the any function for particular duration.
For e.g.:

```
forward();
_delay_ms(3000);
stop();
```

This code snippet will make the robot move in forward direction for 3 seconds.

❖ `pick();`

This function is used to pick the object on the arena. The object should be in the detectable range of the IR sensor to be picked up. Once an object is picked up, you cannot pick up another object until the already picked up object is placed.

❖ `place();`

This function is used to place the object on the arena. The eBot places it 12.5 cms in front of the eBot. You cannot place an object until you have picked one up.

❖ `filter_red();`

This function updates the pulse count after red filter has been applied to the color sensor.

❖ `filter_blue();`

This function updates the pulse count after blue filter has been applied to the color sensor.

❖ `filter_green();`

This function updates the pulse count after green filter has been applied to the color sensor.

❖ `filter_clear();`

This function updates the pulse count after filters have been cleared from the color sensor.