Communication Protocol of the Wheel Sensor Unit

version 1.0

Background

The aim of this document is to define the communication protocol for data exchange between the wheel sensor unit (WSU) and the user application.

The WSU is a special measurement device designed for research purposes with the aim to investigate the possibility of estimating the velocity of a wheeled vehicle from inertial readings captured from its wheels. The main goal of the unit is to provide a fast and simple method for capturing synchronized measurement of angular velocity, linear acceleration and optionally wheel radius changes. The unit design is based on IoT principles, which allows great connectivity and scalability of the system. Due to the specific application, the communication is conducted wirelessly via standard 2,4GHz IEEE 802.11b/g network. The communication protocol is hence text-based in order to achieve the best portability between various systems. Upon startup the WSU tries to connect to preconfigured WiFi network. If the network is not found within approximately 15 seconds, the device switches to AccessPoint (AP) mode, which allows its wireless configuration.

Incoming message

The input of the WSU is the configuration data, used for specifying all parameters of the system. The protocol is based on HTTP/1.1. Incoming messages consist of HTTP header and data. Required fields in the header are:

Field	Description	Example
HTTP operation	Fixed value	POST / HTTP/1.1
Content length	Number of bytes in the body, case insensitive	Content-Length: 125

Data are separated from the header by an empty line containing only CR and LF characters. Data are URL encoded (mime-type application/x-www-form-urlencoded). The POST request updates existing settings. Empty fields are ignored, hence not updated. The available fields are:

Field	Description	C-type	Default value
`dev_id`	Unique device ID	uint32_t	1
`dev_ssid`	SSID of WiFi network to connect to	char[32]	"KRIS_AB205"
`dev_pwd`	New WiFi network password	char[64]	"robot205"
`ap_ssid`	SSID of WiFi in AP mode	char[32]	"WH_IMU1"
`ap_pwd`	New WiFi AP network password	char[64]	"12345678"
`udp_host`	Data stream target IP	char[32]	"192.168.0.1"
`udp_port`	Data stream target port	uint16_t	5001
`ntp_host`	NTP server name	char[32]	"pool.ntp.org"
`ntp_offset`	NTP server offset w. r. t. GMT in hours	uint32_t	1
`ntp_daylight`	Flag signalizing daylight offset in hours	uint32_t	1
`gyro_fs`	Gyroscope full scale in deg/s	uint32_t	2000
`accel_fs`	Accelerometer full scale in g	uint32_t	4
`imu_odr`	IMU output data rate in Hz	float	208.0
`lidar_period`	Laser distance sensors sample period in ms	uint32_t	50
`save`	Flag must be set to non-empty string to save parameters	char[8]	""
`refresh`	Flag triggers WiFi network scan	char[8]	""
`connect`	Flag triggers connection attempt to WiFi	char[8]	""

Note that given field lengths include null termination of strings. The configuration is also available in user-friendly web interface accessible at the URL http://<device IP address>

Outgoing message

Outgoing data stream is UDP. One datagram may contain one or more samples from sensors. The protocol is in text mode, fields are semicolon-separated. Each sample ends with CRLF sequence "\r\n". Each sample contains all the fields in fixed order:

Field index	Description	Python type
0	Device ID	int
1	UNIX Timestamp in seconds	float
2	Temperature in degrees of Celsius	float
3	Gyroscope X-axis in deg/s	float
4	Gyroscope Y-axis in deg/s	float
5	Gyroscope Z-axis in deg/s	float
6	Acclerometer X-axis in g	float
7	Acclerometer Y-axis in g	float
8	Acclerometer Z-axis in g	float
9	Raw value of distance sensor 1	float
10	Raw value of distance sensor 2	float
11	Raw value of distance sensor 3	float
12	Estimated raw value RMS of distance sensor 1	float
13	Estimated raw value RMS of distance sensor 2	float
14	Estimated raw value RMS of distance sensor 3	float

Acknowledgement