Parameter configuration guide of the config.h file

TSDZ2 open source firmware mb.20beta1.B
modified version of 20 beta 1 (C)
adapted to the original VLCD5 - VLCD6 - XH18 displays

I recommend reading the 20 beta 1 instruction manual (wiki).
It was a source of useful information for the preparation of this guide.

This guide replaces that of the previous version.
The added or modified parameters are highlighted in yellow.

Before using the software, modify the parameters according to your needs.
Check the correctness of the type of motor, battery and display.
Caution. Changing only the numeric values after the definitions, consistent with the definitions themselves and in the expected range, incorrect values can cause unpredictable operations

Now you can also change the parameters using the JavaConfigurator software.
The red texts refer to the use of the configurator.

```
/*
 * config.h
 * parameters configuration file
 * for TongSheng TSDZ2 motor controller firmware
 * by (C) Casainho and EndlessCadence and Leon, 20 beta 1 version
 * adapted to stock displays VLCD5 VLCD6 XH18
 * from an idea of marcoq (Jobike forum)
 * Author: mbrusa
 * Version mb.20beta1.B
 */

#ifndef CONFIG_H_
#define CONFIG_H_

// ----------------------------------------------------------------------------------------------------------------------
//  DEBUG
// ----------------------------------------------------------------------------------------------------------------------

#define DEBUG_MODE                                  0
#define DEBUG_DATA                                  0


// ----------------------------------------------------------------------------------------------------------------------
//  MOTOR
// ----------------------------------------------------------------------------------------------------------------------

// Motor type
// motor type (0=48V 1=36V)
// DO NOT ENTER OTHER VALUES!
#define MOTOR_TYPE                                  1
```

Choice of motor type 0 = 48V 1 = 36V. DO NOT ENTER OTHER VALUES!

```
// Experimental high cadence mode
// experimental high cadence mode (1=ENABLED)
#define EXPERIMENTAL_HIGH_CADENCE_MODE              0
```

If enabled, it allows assistance from the motor at a higher cadence. It is possible that the experimental mode is not suitable for the motor, use at your own risk. Probably only suitable for 36 V motors.

// Motor acceleration

```
// motor acceleration adjustment
#define MOTOR_ACCELERATION                                  25
/* VALUES NEED VALIDATION FROM USER FEEDBACK
        Default value = 0 %
        36 volt motor, 36 volt battery = 35 %
        36 volt motor, 48 volt battery = 5 %
        36 volt motor, 52 volt battery = 0 %
        48 volt motor, 36 volt battery = 45 %
        48 volt motor, 48 volt battery = 35 %
        48 volt motor, 52 volt battery = 30 % */
```

Acceleration of the motor. As a first setting, use low values, then gradually increase if necessary.
Consider the values in the table as maximum values.
Set carefully, aware that setting a higher value than necessary can cause greater stress on the transmission.

// Start-up assistance without pedaling

```
// assist without pedal rotation (1=ENABLED)
#define MOTOR_ASSISTANCE_WITHOUT_PEDAL_ROTATION             0
```

At the start, enable assistance only with the push on the pedals without rotation.
It is recommended to use this function with the brake sensors installed and enabled.
The minimum thrust needed to start the assistance is adjusted with the next parameter.

// Assist without pedaling threshold

```
// assist without pedal rotation threshold
// max value 100, recommended range between 10-20
#define ASSISTANCE_WITHOUT_PEDAL_ROTATION_THRESHOLD         20
```

Sensitivity to start assistance without pedal rotation.
As a first setting, use low values, then gradually increase if necessary.
Set to 100% just apply a minimum torque to the pedals. Recommended values 10-20

// Pedal torque ADC step

```
// pedal torque conversion (optional calibration)
#define PEDAL_TORQUE_PER_10_BIT_ADC_STEP_X100               67
```

Conversion factor of the torque applied to the pedal.
It is used for a correct calculation of human power.
With higher values there is a greater reactivity. Gradually change by a few units.
Calibration recommended (see specific instructions).
After calibration, enter the actual value.
It does not replace the hardware calibration, necessary to improve the resolution of the torque sensor.

// Pedal torque ADC range

```
// pedal torque actual range (calibration required)
#define PEDAL_TORQUE_10_BIT_ADC_RANGE                       160
```

Actual value of the maximum torque applied to the pedal.
It is the difference between the adc value with maximum torque and the adc value without pushing on the pedals.
It is used to amplify the range of use of the torque sensor when too limited.
The actual value can only be obtained with a correct calibration (see specific instructions).
After calibration, enter the actual value. Maximum 255.

// Pedal torque ADC offset adjustment

```
// adc torque offset adjustment
#define ADC_TORQUE_OFFSET_ADJUSTMENT                    0
```

Parameter for adjusting the ADC offset of the torque sensor.
The value is subtracted from the one calculated at power-on, it is used to give greater sensitivity to the push on the pedals.
Values from 0 to 20 max.
Attention. Too high values can cause an unwanted start and / or a delayed motor stop.

// Cadence sensor high percentage

```
// cadence sensor high percentage (calibration required)
#define CADENCE_SENSOR_PULSE_HIGH_PERCENTAGE_X10        500
```

Parameter for cadence sensor in advanced mode. Default value.
The actual value can only be obtained with a correct calibration (see specific instructions).
See the procedure on the dedicated wiki page (20 beta 1).
After calibration, enter the actual value.

// Motor blocked error

```
// ERROR_MOTOR_BLOCKED parameter
#define MOTOR_BLOCKED_COUNTER_THRESHOLD                 2    // from 1 to 10 (0.1 second)
#define MOTOR_BLOCKED_BATTERY_CURRENT_THRESHOLD_X10     30   // from 10 to 50 (Amp x 10)
#define MOTOR_BLOCKED_ERPS_THRESHOLD                    20   // from 10 to 20 (ERPS)
```

ERROR_MOTOR_BLOCKED parameters for blocked motor or wheel control.
They have been moved to config.h to be able to easily modify them according to your needs.
The default values are for immediate intervention and to preserve the blue gear.
Change them in case of unwanted interventions.
MOTOR_BLOCKED_COUNTER_THRESHOLD, intervention time from 1 to 10 (1 = 0.1 seconds)
MOTOR_BLOCKED_BATTERY_CURRENT_THRESHOLD_X10, current from 10 to 50 (30 = 3 Amp)
MOTOR_BLOCKED_ERPS_THRESHOLD, number of revolutions below which the motor is considered to be stopped. Higher values anticipate the intervention, from 10 to 20 ERPS.

// Motor ramp down additional

```
// duty cycle ramp down additional (0 to 20)
#define PWM_DUTY_CYCLE_RAMP_DOWN_MIN_ADDITIONAL     0
```

Additional motor deceleration ramp.
With the "fix overrun" function, the motor stop may be too abrupt, gradually increase.
This parameter also has a second function, with a value greater than zero, it also increases the control time of the stationary pedals, it is a remedy for a possible lack of assistance at startup, due to "fix overrun".
With brake sensors installed, increase only if necessary and by a few units.

```
// ------------------------------------------------------------------------------------------------------------------------
//  BATTERY
// ------------------------------------------------------------------------------------------------------------------------
```

// Battery current max (A)

```
// maximum battery current (Amp)
#define BATTERY_CURRENT_MAX                             16
```

Maximum battery current in amperes.
Set the maximum current that the battery can supply.
The maximum value is internally limited by the software to the value of 18 A.

// Battery power max (W)
// maximum battery power (Watt)
#define TARGET_MAX_BATTERY_POWER 500
Maximum power supplied by the battery in watts.
It is also the motor power limit in OFFROAD mode.

// Battery capacity (Wh)
// battery capacity (Wattora)
#define TARGET_MAX_BATTERY_CAPACITY 630
Total battery capacity in Watt hours.
Calculate the capacity by multiplying the rated voltage by Ah.
Example: a 36 Volt, 17.5 Ah battery has a nominal capacity of 630 Wh.

// Battery cells number
// number of cells in series (7=24V 10=36V 13=48V 14=52V)
#define BATTERY_CELLS_NUMBER 10
Number of battery cells in series.
This value can be an integer from 7 to 14. 7 for 24 V battery; 10 36 V; 13 48 V; 14 of 52 V.

// Battery resistance (milliOhms)
// battery internal resistance (milliohms)
#define BATTERY_PACK_RESISTANCE 196
Internal resistance of the battery in milliohms. It is used to eliminate the oscillation of the notches indicating the state of charge of the battery, between no-load voltage and voltage under load. How to calculate the battery resistance: Measure the difference between the no-load voltage and under load, with a constant current. Example for 10 A current, R = difference of 1.96 Volts / 10 Amp, R = 0.196 = 196 milliohm.

// Battery voltage cut off (V)
// battery low-cut-off voltage (Volt)
#define BATTERY_LOW_VOLTAGE_CUT_OFF 29
Cut-off voltage for low battery. If the voltage drops below this value, the controller will automatically lower the current so as not to drop below the minimum voltage limit. Set this value by checking the characteristics of the cells from the battery.

// Battery voltage calibration (%)
// battery voltage calibration (95% to 105%)
#define ACTUAL_BATTERY_VOLTAGE_PERCENT 100
Parameter to correct the voltage value shown on the display.
Example, with a fully charged battery of nominal 36V, the voltage should be close to 42V, if it is lower try to increase the parameter one unit at a time until reading 42V, vice versa if the voltage is higher, the parameter must be decreased.

// Battery capacity calibration (%)
// battery capacity calibration (max 100%)
#define ACTUAL_BATTERY_CAPACITY_PERCENT 100
Parameter for setting the actual battery capacity.
Calibration procedure: With the battery fully charged, check the percentage on the display, it must be 99.9%. At this point, use the bike until the battery is completely exhausted.
Check the residual percentage and calculate the effective percentage value (100 - residual value). Set the parameter with this value. Example, final residual percentage 8%, effective capacity 92% (100 - 8).

// li-ion cell
// Overvoltage (V)
#define LI_ION_CELL_OVERVOLT 4.35
Value beyond which the error E08-ERROR_OVERVOLTAGE is displayed.
Possible if you set the wrong number of cells in series.
The unit of measurement of this and subsequent parameters is in Volts (of each single cell).

// Reset SOC percentage (V)
#define LI_ION_CELL_RESET_SOC_PERCENT 4.05
Value for automatic reset to 99.9% of the percentage of residual capacity, with battery fully charged.
Recommended values from 4.10 to 4.15, otherwise with lower values, after a short turn, if the voltage does not fall below this value, when it is switched on again it resets to 99.9 again. If the battery is not fully charged and the voltage is lower than this value, the reset is not automatically activated. If desired, it can be done manually by activating the procedure provided.
// full
// Cell voltage full (V)
#define LI_ION_CELL_VOLTS_FULL 3.95
Minimum voltage value to display the complete battery charge status, 4 notches with VLCD6 and XH18 display, 6 notches with VLCD5.

// 4 bars
// Cell voltage 3/4 (V)
#define LI_ION_CELL_VOLTS_3_OF_4 3.85
// Cell voltage 2/4 (V)
#define LI_ION_CELL_VOLTS_2_OF_4 3.55
// Cell voltage 1/4 (V)
#define LI_ION_CELL_VOLTS_1_OF_4 3.25
Voltage value to display intermediate charge states.
From 1 to 3 notches, for VLCD6 and XH18 displays.

// 6 bars
// Cell voltage 5/6 (V)
#define LI_ION_CELL_VOLTS_5_OF_6 3.80
// Cell voltage 4/6 (V)
#define LI_ION_CELL_VOLTS_4_OF_6 3.65
// Cell voltage 3/6 (V)
#define LI_ION_CELL_VOLTS_3_OF_6 3.50
// Cell voltage 2/6 (V)
#define LI_ION_CELL_VOLTS_2_OF_6 3.25
// Cell voltage 1/6 (V)
#define LI_ION_CELL_VOLTS_1_OF_6 3.10
Voltage value to display intermediate charge states.
From 1 to 5 notches, for VLCD5 display.

// empty
//Cell voltage empty (V)
#define LI_ION_CELL_VOLTS_EMPTY 2.90
Voltage value to display the state of the completely discharged battery, 0 notches.
For all these parameters, check the technical characteristics of the cells used.

```
// ------------------------------------------------------------------------------------------------------------------------
//  BIKE
// ------------------------------------------------------------------------------------------------------------------------
```

// Wheel circumference (mm)

```
// wheel perimeter(mm)
#define WHEEL_PERIMETER                                 2280
```

This parameter is used to calculate the speed and kilometers traveled.
Enter the wheel perimeter in millimeters.
Indicative values:
26-inch wheel = 2050 mm
27-inch wheel = 2150 mm
27.5 inch wheel = 2215 mm
28-inch wheel = 2250 mm
29-inch wheel = 2300 mm
It is recommended to measure the actual perimeter and check the distance traveled with GPS.

// Max speed (km/h)

```
// wheel max speed (km/h)
#define WHEEL_MAX_SPEED                                 45
```

Maximum speed limit. Beyond this value the motor stops.
Attention, if the ENABLE_WHEEL_MAX_SPEED_FROM_DISPLAY function is enabled, this limit is ignored and replaced by the one set on the display.

```
// ------------------------------------------------------------------------------------------------------------------------
//  FUNCTION
// ------------------------------------------------------------------------------------------------------------------------

// enable functions (1=ENABLED)
```

// Lights

```
#define ENABLE_LIGHTS                                   1
```

Enable the use of lights, on and off, via the lights button.

// Walk assist

```
#define ENABLE_WALK_ASSIST                              1
```

Enables the use of the walk assist, walking the bike up to 6 km / h.

// Brake sensor

```
#define ENABLE_BRAKE_SENSOR                             0
```

Enable the use of brake sensors when installed.
It also enables the functions where the use of sensors is required:
- walk assist debounce delay
- cruise mode without pedal movement
- accelerator
For safety with the sensors installed, even with the function disabled, the motor stop is always active when braking.

// ADC optional disabile (not Trottle and not Temperature sensor)

// Throttle

```
#define ENABLE_THROTTLE                                 0
```

Enable the accelerator only if it has been installed.

Only with brake sensors installed and enabled.
Inquire about legislative restrictions regarding use.
Attention, the accelerator is an alternative to the temperature sensor.
Both cannot be enabled!

// Temperature sensor
#define ENABLE_TEMPERATURE_LIMIT 0
Enable only if the temperature sensor has been installed.
Warning, the temperature sensor is an alternative to the accelerator.
Both cannot be enabled!

// FUNCTIONS ENABLED ON STARTUP
// Street mode on startup
// street mode (0=OFFROAD 1=STREET)
#define ENABLE_STREET_MODE_ON_STARTUP 1
Enable STREET mode at startup.
The STREET mode is a function that can be configured as a legal driving mode, it is possible to limit the speed and power of the motor.
The throttle and cruise mode are disabled.
Inquire about legislative restrictions regarding speed limit and power limit.

// Set parameters on startup
// display mode (0=DISPLAY DATA 1=SET PARAMETER)
#define ENABLE_SET_PARAMETER_ON_STARTUP 0
Choice of how to use the display.
If enabled, parameter modification is active at startup.
If left at 0, the data display is active at startup.

// Odometer compensation
// odometer compensation (1=ENABLED)
#define ENABLE_ODOMETER_COMPENSATION 1
Enable the compensation of the kilometers added when viewing the data.
Even when the bike is stationary, all data sent to the display increase the odometer.
By enabling this function, the distance added and not traveled is recovered, during this operation the speed displayed while driving remains at zero until the kilometers have been balanced.

// Cadence sensor adv. on startup
// cadence sensor mode (0=STANDARD 1=ADVANCED)
#define CADENCE_SENSOR_MODE_ON_STARTUP 0
Cadence sensor mode at startup.
0 = standard mode
1 = advanced mode (double the pulses).
Choosing advanced mode requires a calibration (see specific instructions).

// Torque sensor adv. on startup
// torque sensor mode (0=STANDARD 1=ADVANCED)
#define TORQUE_SENSOR_MODE_ON_STARTUP 0
Torque sensor mode at startup.
0 = standard mode
1 = advanced mode (linearization not yet implemented).
A calibration is possible in each mode (see specific instructions).

// Lights mode on startup
// lights configuration (0 to 8)
#define LIGHTS_CONFIGURATION_ON_STARTUP 0

Mode of operation of the lights at startup.
See below the various modes and their codes.
This value can be different from the 3 selectable in the display menu.

// Enable on startup (Power assist, Torque assist, Cadence assist, eMTB assist)
// ridind mode (1=POWER 2=TORQUE 3=CADENCE 4=EMTB)
#define RIDING_MODE_ON_STARTUP 1

Start-up assistance mode.
Choose the preferred one from the available assistance modes.
1 - POWER assistance proportional to the power on the pedals
2 - TORQUE assistance proportional to the torque on the pedals
3 - CADENCE assistance subordinated to the movement of the pedals
4 - EMTB assistance with progressive percentage of the torque on the pedals

// lights configuration
// Lights mode 1
#define LIGHTS_CONFIGURATION_1 1
// Lights mode 1
#define LIGHTS_CONFIGURATION_2 6
// Lights mode 1
#define LIGHTS_CONFIGURATION_3 7
/ NOTE: regarding the various light modes*
(0) lights ON when enabled
(1) lights FLASHING when enabled
(2) lights ON when enabled and BRAKE-FLASHING when braking
(3) lights FLASHING when enabled and ON when braking
(4) lights FLASHING when enabled and BRAKE-FLASHING when braking
(5) lights ON when enabled, but ON when braking regardless if lights are enabled
(6) lights ON when enabled, but BRAKE-FLASHING when braking regardless if lights are enabled
(7) lights FLASHING when enabled, but ON when braking regardless if lights are enabled
(8) lights FLASHING when enabled, but BRAKE-FLASHING when braking regardless if lights are enabled
*/

Configuration of the light modes, selectable from the menu on the display.
Find out about compliance with current regulations.
Choose the 3 preferred modes among the 9 available. With light control ON:
0 - on
1 - flashing
2 - on and fast flashing when braking
3 - flashing and on when braking
4 - flashing and fast flashing when braking
5 - on and on during braking also with light control OFF
6 - on and fast flashing when braking even with the light control OFF
7 - flashing and switched on when braking even with the light control OFF
8 - flashing and fast flashing when braking even with the light control OFF

The braking modes are only available with the brake sensors installed.

```
// STREET MODE FUNCTION
// Street power limit enebled
// street mode power limit (1=ENABLED)
#define STREET_MODE_POWER_LIMIT_ENABLED 1
```
Enable the power limit in STREET mode.

```
// Street power limit (W)
// street mode power limit value (Watt)
#define STREET_MODE_POWER_LIMIT 500
```
Power limit in watts when STREET mode is enabled.

```
// Street speed limit (km/h)
// street mode speed limit (km/h)
#define STREET_MODE_SPEED_LIMIT 25
```
Speed limit in km / h when STREET mode is enabled.
Beyond this value the motor stops.
This speed limit can be replaced by the one set on the display when the ENABLE_WHEEL_MAX_SPEED_FROM_DISPLAY function is enabled.

```
// street mode enable other functions
// Throttle on street mode
#define STREET_MODE_THROTTLE_ENABLED 0
```
Enable the accelerator, if installed, in STREET mode.
Available only with mounted and enabled brake sensors.

```
// Cruise on street mode
#define STREET_MODE_CRUISE_ENABLED 0
```
Enable cruise mode in STREET mode.
Available in mode with active pedal movement, otherwise only with brake sensors installed and enabled.

```
// throttle ADC values (optional)
// ADC throttle min value
#define ADC_THROTTLE_MIN_VALUE 47
// ADC throttle max value
#define ADC_THROTTLE_MAX_VALUE 176
```
Throttle adjustment range, relative minimum and maximum values.

```
// motor temperature limit (sensor required)
// Motor temperature min limit
#define MOTOR_TEMPERATURE_MIN_VALUE_LIMIT 65
```
Set the temperature from which the motor protection will start, limiting the power. The power gradually decreases up to the maximum temperature limit, then the motor stops.

```
// Motor temperature max limit
#define MOTOR_TEMPERATURE_MAX_VALUE_LIMIT 80
```
Set the maximum motor temperature. At this temperature the motor will be turned off. Values in degrees Celcius.

```
// Temperature error with min limit
// enable temperature error at min limit value
#define ENABLE_TEMPERATURE_ERROR_MIN_LIMIT 1
```
If enabled, the error code E06 - ERROR_OVERTEMPERATURE is displayed when the minimum temperature limit is exceeded. If disabled when the maximum limit is exceeded.

```
// ------------------------------------------------------------------------------------------------------------------------------
//  DISPLAY
// ------------------------------------------------------------------------------------------------------------------------------
```

// Display type

```
// display type  (1=ENABLED)
#define ENABLE_VLCD6                    0
#define ENABLE_VLCD5                    0
#define ENABLE_XH18                     1
```

Choose the type of display used. Enable only one.

```
// display parameters
```

// Display working on

```
#define ENABLE_DISPLAY_WORKING_FLAG     1
```

Enables the display to turn off after 5 minutes of inactivity.

// Display always on

```
#define ENABLE_DISPLAY_ALWAYS_ON        0
```

The display is always on. This parameter is alternative to the previous one, enable only one of the two.

// Set max speed from display

```
#define ENABLE_WHEEL_MAX_SPEED_FROM_DISPLAY     0
```

Enable the speed limit set on the display.
The upper limit of the WHEEL_MAX_SPEED parameter is ignored.
The speed limit always remains active in STREET mode.
Attention, when the speed limit on the display is lower than that in STREET mode, the one on the display has priority.
Example:
- display limit 30 km / h, STREET limit 25 km / h, limit used 25 km / h
- display limit 20 km / h, STREET limit 25 km / h, limit used 20 km / h

For setting the limit, consult the manual of your display.

// Time to menu items (0.1 s)

```
// delay menu function (0.1 sec, 40 to 60)
#define DELAY_MENU_ON                   50
```

Maximum delay between the first press of the light button and the second confirmation, in the parameter setting procedure.
It is also the time within which, after confirmation and with a flashing code, you can move on to the next parameter.

// Return to default display mode

```
// enable delay return to default display mode
#define ENABLE_RETURN_DEFAULT_DISPLAY_MODE      1
```

Enables automatic return to the display's default use mode.
If disabled, the return to the previous mode must be done manually.

// Time to return default display mode (s)

```
// delay return to default display mode (seconds)
#define DELAY_DISPLAY_MODE_DEFAULT      30
```

Delay after changing the use mode of the display for automatic return to the default mode.
- ENABLE_SET_PARAMETER_ON_STARTUP = 0 it returns to the data display.
- ENABLE_SET_PARAMETER_ON_STARTUP = 1 it returns to the modification of the parameters.

// Auto display data with lights on
// enable auto display data with lights on
#define ENABLE_AUTO_DATA_DISPLAY 1
Enables the automatic display of data in sequence, when the lights are turned on.
The display time is that set for each individual data.

// Number of data auto displayed
// number of data auto displayed
#define AUTO_DATA_NUMBER_DISPLAY 3
Number of data displayed automatically when the lights are turned on.
Value from 1 to 3, or from 1 to 6 if ENABLE_DISPLAY_DOUBLE_DATA is enabled.

// Display second serie data (4 to 6)
// displays double data (0=3 VALUES 1=6 VALUES)
#define ENABLE_DISPLAY_DOUBLE_DATA 0
Enables the display of two data series (3 + 3 values)

// delay display data (0.1 sec, max 255)
// first series
// Time to displayed data 1 (0.1 s)
#define DELAY_DISPLAY_DATA_1 50
// Time to displayed data 2 (0.1 s)
#define DELAY_DISPLAY_DATA_2 50
// Time to displayed data 3 (0.1 s)
#define DELAY_DISPLAY_DATA_3 250

// second series
// Time to displayed data 4 (0.1 s)
#define DELAY_DISPLAY_DATA_4 250
// Time to displayed data 5 (0.1 s)
#define DELAY_DISPLAY_DATA_5 50
// Time to displayed data 6 (0.1 s)
#define DELAY_DISPLAY_DATA_6 50
After the first press of the lights button and the second confirmation, it is the display time of the chosen data, within this time, you can press the lights button again to go to the next data.
The maximum value is 255 = 25.5 seconds and can be different for each individual data.
To stop displaying the data before the end of the time, change the level.

// display data configuration
// first series
// Data 1
#define DISPLAY_DATA_1 1
// Data 2
#define DISPLAY_DATA_2 2
// Data 3
#define DISPLAY_DATA_3 5
// second series
// Data 4
#define DISPLAY_DATA_4 4
// Data 5
#define DISPLAY_DATA_5 7
// Data 6
#define DISPLAY_DATA_6 0

```
/*  display data code configuration
        0 - motor temperature (°C)
        1 - battery SOC remaining (%)
        2 - battery voltage (Volt)
        3 - battery current (Amp)
        4 - absorbed motor power (Watt/10)
        5 - adc torque sensor (8 bit)
        6 - adc torque sensor (10 bit)
        7 - pedal cadence (rpm)
        8 - human power
        9 - cadence sensor pulse high percentage
       10 - pedal weight
       11 - pedal torque adc conversion
       12 - pedal torque adc range
*/
```

Configuration of the data shown on the display.
Choose the data to be displayed, in the type and in the preferred order among those available.
0 – motor temperature, only with sensor installed (° C)
1 - remaining battery charge (%)
2 - battery voltage (Volt)
3 - battery current (Amp)
4 - power absorbed by the motor (Watt / 10)
5 - torque sensor adc value (8 bit)
6 - torque sensor adc value (10 bit)
7 - pedal cadence (rpm)
8 - human power (Watt / 10)
9 - cadence sensor pulse high percentage
10 - pedal weight for calibration (Kg)
11 - pedal torque adc conversion
12 - pedal torque adc range

```
// ----------------------------------------------------------------------------------------------------------------------
//  ASSIST
// ----------------------------------------------------------------------------------------------------------------------
```

CAUTION. For safety, all assistance values at level 0-OFF must be set to zero. 4 gear levels are available for each assistance mode: 1-ECO, 2-TOUR, 3-SPORT, 4-TURBO.

```
// Power assist mode
// power assist (% max 500)
#define POWER_ASSIST_LEVEL_1                    70
#define POWER_ASSIST_LEVEL_2                    120
#define POWER_ASSIST_LEVEL_3                    210
#define POWER_ASSIST_LEVEL_4                    300
```

POWER ASSIST. Assistance mode proportional to the power on the pedals.
Values in percentage, maximum 500%.
Example, applying 100 watts to the pedals, with 300% assistance, the motor delivers 300 watts.

// Torque assist mode

```
// torque assist (max 254)
#define TORQUE_ASSIST_LEVEL_1                70
#define TORQUE_ASSIST_LEVEL_2                100
#define TORQUE_ASSIST_LEVEL_3                130
#define TORQUE_ASSIST_LEVEL_4                160
```

TORQUE ASSIST. Assistance mode proportional to the torque on the pedals.
The power supplied by the motor is proportional to the applied torque and the set assistance values.
Relative values, maximum 254.

// Cadence assist mode

```
// cadence assist (max 254)
#define CADENCE_ASSIST_LEVEL_1               70
#define CADENCE_ASSIST_LEVEL_2               100
#define CADENCE_ASSIST_LEVEL_3               130
#define CADENCE_ASSIST_LEVEL_4               160
```

CADENCE ASSIST. Assistance mode subject to pedal movement.
The power supplied by the motor depends partly on the assistance values set and partly on the cadence of the pedals.
Relative values, maximum 254.

// eMTB assist mode

```
// eMTB assist (sensitivity 0 to 20)
#define EMTB_ASSIST_LEVEL_1                  6
#define EMTB_ASSIST_LEVEL_2                  9
#define EMTB_ASSIST_LEVEL_3                  12
#define EMTB_ASSIST_LEVEL_4                  15
```

EMTB ASSIST. Assistance mode with progressive percentage of the torque on the pedals.
The power supplied by the motor is progressively proportional to the applied torque. 20 predefined sensitivities are available.
Higher values correspond to a more reactive assistance, faster to reach the maximum motor power.
Choose your preferred sensitivity values from those available, from 1 to 20.

// Walk assist mode

```
// walk assist (max 80)
#define WALK_ASSIST_LEVEL_1                  30
#define WALK_ASSIST_LEVEL_2                  40
#define WALK_ASSIST_LEVEL_3                  50
#define WALK_ASSIST_LEVEL_4                  60
```

WALK ASSIST. Assistance mode when you want to accompany the bike on foot up to 6 km / h.
Activated with the dedicated button, consult the manual of your display.
Maximum value 100. Try low values and gradually increase.
Using low gears, high gears stress the transmission.

// Walk assist speed limit

```
// walk assist threshold (speed limit max km/h)
#define WALK_ASSIST_THRESHOLD_SPEED          6
```

Maximum speed limit in walk assist mode, in km / h.
Inquire about legislative restrictions regarding the limit.

// Walk assist debounce time

```
// walk assist debounce (brake sensor required)
#define WALK_ASSIST_DEBOUNCE_ENABLED                    0
```

Enable the debounce time on the walk assist button.
Useful on rough terrain, when a rebound can cause the button to be released unwanted.
Available only with brake sensors installed and enabled.

// Walk assist deb. time

```
// walk assist debounce time (0.1 sec)
#define WALK_ASSIST_DEBOUNCE_TIME                       50
```

Debounce time value on the walk assist button.
It is recommended to set this time as low as possible, slightly higher than that necessary for the activation of the walk assist.
Attention, the assistance remains active after releasing the button for the set time.
To stop assistance during this time, change the level.
With display XH18 and VLCD5 it stops only when moving to the upper level.

// Cruise mode

```
// cruise level (target km/h, brake sensor required)
#define CRUISE_TARGET_SPEED_LEVEL_1                     15
#define CRUISE_TARGET_SPEED_LEVEL_2                     18
#define CRUISE_TARGET_SPEED_LEVEL_3                     21
#define CRUISE_TARGET_SPEED_LEVEL_4                     24
```

CRUISE ASSIST. Assistance mode with speed control.
The value set in the levels is the target speed to be maintained in km / h.
The power supplied by the motor adjusts itself to maintain the chosen speed. speed may not be reached due to limited motor power.
The speed limits seen above have priority.
Carefully read the function of the next parameter.

// Cruise without pedaling

```
// cruise function without pedal rotation
#define CRUISE_MODE_WALK_ENABLED                        0
```

Set to 0
The cruise activation mode is subordinated to the movement of the pedals.
Speed is maintained only with minimal pedal movement.
By stopping pedaling the motor stops. It is the default mode.
It can be compared to the CADENCE ASSIST mode, the difference is that changing the level does not change the power supplied by the motor but the speed to be achieved.

Set to 1
You can maintain your speed even without turning the pedals by pressing the walk assist button. By releasing the button and stopping pedaling, the motor stops.
It can be compared to an accelerator with speed variation in steps. Increasing the level increases the speed, decreasing it decreases, at 0-OFF the motor stops.
Available only with brake sensors installed and enabled.
Inquire about legislative restrictions regarding use.
In STREET mode with cruise mode disabled (STREET_MODE_CRUISE_ENABLED at 0), assistance is activated in CADENCE ASSIST mode.

```
// Speed cruise enabled
// cruise threshold (speed limit min km/h)
#define CRUISE_THRESHOLD_SPEED                    10
```
Minimum speed limit for activating cruise mode, in km / h.
Below this value, assistance in CADENCE ASSIST mode is active.

```
#endif // CONFIG_H_
```