

Other Service API specification

**Note/
CWORDXX is mask word for confidential information.**

2019.10.17
TOYOTA MOTOR CORPRATION

Module Documentation

BaseSystem

[Other service](#)

Detailed Description

Other_service

[Vehicle parameter library](#)

[Event library](#)

[Posix based os001 legacy library](#)

[Rpc library](#)

Detailed Description

Vehicle_parameter_library

```
#define VP\_CWORD63\_VERSION "VP_CWORD63_version"  
#define VP\_CWORD63\_CWORD63\_ID "VP_CWORD63__CWORD63_ID"  
#define VP\_CWORD63\_MAKER\_ID "VP_CWORD63_maker_ID"  
#define VP\_CWORD63\_AREA "VP_CWORD63_area"  
#define VP\_CWORD63\_GRADE "VP_CWORD63_grade"  
#define VP\_CWORD63\_MODEL\_ID "VP_CWORD63_model_ID"  
#define VP\_CWORD63\_DEST\_C\_CODE "VP_CWORD63_dest_c_code"  
#define VP\_CWORD63\_RADIO\_TUNER\_NUM "VP_CWORD63_radio_tuner_num"  
#define VP\_CWORD63\_RADIO\_ANTENNA\_NUM "VP_CWORD63_radio_antenna_num"  
#define VP\_CWORD63\_DAB\_TUNER\_NUM "VP_CWORD63_DAB_tuner_num"  
#define VP\_CWORD63\_DAB\_SUB\_ANTENNA "VP_CWORD63_DAB_sub_antenna"  
#define VP\_CWORD63\_CWORD32\_DAB "VP_CWORD63__CWORD32_DAB"  
#define VP\_CWORD63\_CACHERADIO "VP_CWORD63_CacheRadio"  
#define VP\_CWORD63\_AMP\_TYPE "VP_CWORD63_AMP_type"  
#define VP\_CWORD63\_OP\_COOPERATION "VP_CWORD63_OP_cooperation"  
#define VP\_CWORD63\_DISPLAY "VP_CWORD63_DISPLAY"  
#define VP\_CWORD63\_TOUCHPANEL "VP_CWORD63_TOUCHPANEL"  
#define VP\_CWORD63\_DECK "VP_CWORD63_DECK"  
#define VP\_CWORD63\_CWORD105 "VP_CWORD63__CWORD105_"  
#define VP\_CWORD63\_DEFAULT\_MODE "VP_CWORD63_default_mode"
```

```

#define
    VP_CWORD63_VOICE_RECOGNITION_TUTORIAL "VP_CWORD63_VOICE_RECOGNITION_T
    UTORIAL"
#define VP_CWORD63_SEARCH "VP_CWORD63_SEARCH"
#define VP_CWORD63_CONTACT "VP_CWORD63_CONTACT"
#define VP_CWORD63_JPTRAFFICINFO_SHOW "VP_CWORD63_JPTrafficInfo_show"
#define
    VP_CWORD63_INFOMATION_MANAGEMENT "VP_CWORD63_INFOMATION_MANAGEME
    NT"
#define VP_CWORD63_SUBSCREEN_APPS_SW "VP_CWORD63_SUBSCREEN_APPS_SW"
#define VP_CWORD63_WIFI_STA_ENABLE "VP_CWORD63_WIFI_STA_ENABLE"
#define VP_CWORD63_WIFI_AP_ENABLE "VP_CWORD63_WIFI_AP_ENABLE"
#define VP_CWORD63_WIFI_DIRECT_ENABLE "VP_CWORD63_WIFI_DIRECT_ENABLE"
#define VP_CWORD63_HAS_GPS "VP_CWORD63_HAS_GPS"
#define
    VP_CWORD63_HAS_CWORD80_DEFAULT "VP_CWORD63_HAS_CWORD80_DEFAULT"
#define
    VP_CWORD63_DISCONNECT_CWORD80_TIMEOUT "VP_CWORD63_DISCONNECT_CW
    ORD80_TIMEOUT"
#define VP_CWORD63_PHASE "VP_CWORD63_PHASE"
#define VP_CWORD63_CERTIFICATION_SOFT "VP_CWORD63_CERTIFICATION_SOFT"
#define VP_CWORD63_CWORD6 "VP_CWORD63_CWORD6_"
#define VP_CWORD63_VOICE_TAGS "VP_CWORD63_VOICE_TAGS"
#define VP_CWORD63_BT_2HFP "VP_CWORD63_BT_2HFP"
#define VP_CWORD63_CONNECTION_SERVER "VP_CWORD63_CONNECTION_SERVER"
#define
    VP_CWORD63_CWORD33_LOG_ENC_KEYTYPE "VP_CWORD63_CWORD33_LOG_ENC_KE
    YTYPE"
#define VP_CWORD63_DES_VOL_1 "VP_CWORD63_DES_VOL_1"
#define VP_CWORD63_DES_VOL_2 "VP_CWORD63_DES_VOL_2"
#define VP_CWORD63_DES_VOL_3 "VP_CWORD63_DES_VOL_3"
#define VP_CWORD63_DES_VOL_4 "VP_CWORD63_DES_VOL_4"
#define VP_CWORD63_DES_VOL_5 "VP_CWORD63_DES_VOL_5"
#define VP_CWORD63_DES_VOL_6 "VP_CWORD63_DES_VOL_6"
#define VP_CWORD63_DES_VOL_7 "VP_CWORD63_DES_VOL_7"
#define VP_CWORD63_DES_VOL_8 "VP_CWORD63_DES_VOL_8"
#define VP_CWORD63_STEERING_TYPE "VP_CWORD63_STEERING_TYPE"
#define
    VP_CWORD63_HANDWRITEMODE_MOVERANGE "VP_CWORD63_Handwritemode_mover
    ange"
#define VP_CWORD63_TOUCHPAD_HOME_SW "VP_CWORD63_TOUCHPAD_HOME_SW"
#define VP_CWORD63_CWORD17 "VP_CWORD63_CWORD17_"
#define VP_CWORD63_BT_ADD_DEVICE_SW "VP_CWORD63_BT_ADD_DEVICE_SW"
#define
    VP_CWORD55_CWORD66_SVR_PRODUCT_DOWNLOAD_URL "VP_CWORD55_CWORD6
    6_SVR_PRODUCT_DOWNLOAD_URL"

```

```

#define
  VP_CWORD55_CWORD66_SVR_TRY_DOWNLOAD_URL "VP_CWORD55_CWORD66_SV
  R_TRY_DOWNLOAD_URL"
#define
  VP_CWORD55_CWORD66_SVR_TEST_DOWNLOAD_URL "VP_CWORD55_CWORD66_S
  VR_TEST_DOWNLOAD_URL"
#define
  VP_CWORD55_CWORD66_SVR_PRODUCT_UPLOAD_URL "VP_CWORD55_CWORD66_S
  VR_PRODUCT_UPLOAD_URL"
#define
  VP_CWORD55_CWORD66_SVR_TRY_UPLOAD_URL "VP_CWORD55_CWORD66_SVR_TR
  Y_UPLOAD_URL"
#define
  VP_CWORD55_CWORD66_SVR_TEST_UPLOAD_URL "VP_CWORD55_CWORD66_SVR_T
  EST_UPLOAD_URL"
#define
  VP_CWORD55_CWORD66_SVR_PRODUCT_BROWSER_URL "VP_CWORD55_CWORD66_
  _SVR_PRODUCT_BROWSER_URL"
#define
  VP_CWORD55_CWORD66_SVR_TRY_BROWSER_URL "VP_CWORD55_CWORD66_SVR_
  TRY_BROWSER_URL"
#define
  VP_CWORD55_CWORD66_SVR_TEST_BROWSER_URL "VP_CWORD55_CWORD66_SVR_
  TEST_BROWSER_URL"
#define
  VP_CWORD55_CWORD66_SVR_PRODUCT_DOWNLOAD_HOST "VP_CWORD55_CWOR
  D66_SVR_PRODUCT_DOWNLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_TRY_DOWNLOAD_HOST "VP_CWORD55_CWORD66_S
  VR_TRY_DOWNLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_TEST_DOWNLOAD_HOST "VP_CWORD55_CWORD66_
  SVR_TEST_DOWNLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_PRODUCT_UPLOAD_HOST "VP_CWORD55_CWORD66_
  _SVR_PRODUCT_UPLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_TRY_UPLOAD_HOST "VP_CWORD55_CWORD66_SVR_
  TRY_UPLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_TEST_UPLOAD_HOST "VP_CWORD55_CWORD66_SVR
  _TEST_UPLOAD_HOST"
#define
  VP_CWORD55_REPRO_SVR_PRODUCT_URL "VP_CWORD55_REPRO_SVR_PRODUCT_URL"
#define VP_CWORD55_REPRO_SVR_TRY_URL "VP_CWORD55_REPRO_SVR_TRY_URL"
#define VP_CWORD55_REPRO_SVR_TEST_URL "VP_CWORD55_REPRO_SVR_TEST_URL"
#define VP_CWORD63_CWORD70_EU "VP_CWORD63_CWORD70_EU"
#define VP_CWORD63_CWORD40_ "VP_CWORD63_CWORD40_"

```

```

#define VP_CWORD63_DCM_EC "VP_CWORD63_DCM_EC"
#define
    VP_CWORD63_EXTERNALDISPLAY_INTERACTION "VP_CWORD63_ExternalDisplay_Interaction"
#define VP_CWORD63_DES_OUTPUT_FL_SP "VP_CWORD63_DES_OUTPUT_FL_SP"
#define VP_CWORD63_DES_OUTPUT_FR_SP "VP_CWORD63_DES_OUTPUT_FR_SP"
#define VP_CWORD63_DES_OUTPUT_RL_SP "VP_CWORD63_DES_OUTPUT_RL_SP"
#define VP_CWORD63_DES_OUTPUT_RR_SP "VP_CWORD63_DES_OUTPUT_RR_SP"
#define VP_CWORD63_DES_OUTPUT_FC_SP "VP_CWORD63_DES_OUTPUT_FC_SP"
#define VP_CWORD63_DES_OUTPUT_ML_SP "VP_CWORD63_DES_OUTPUT_ML_SP"
#define VP_CWORD63_DES_OUTPUT_MR_SP "VP_CWORD63_DES_OUTPUT_MR_SP"
#define VP_CWORD63_DES_OUTPUT_NAVI_SP "VP_CWORD63_DES_OUTPUT_NAVI_SP"
#define VP_CWORD63_DES_OUTPUT_HRDRIVER "VP_CWORD63_DES_OUTPUT_HRDriver"
#define
    VP_CWORD63_DES_OUTPUT_HRPASSENGER "VP_CWORD63_DES_OUTPUT_HRPassenger"
#define VP_CWORD63_AUTOMATED_DRIVING "VP_CWORD63_AUTOMATED_DRIVING"
#define VP_CWORD63_OPN_MOV "VP_CWORD63_OPN_MOV"
#define VP_CWORD63_CWORD84_SDSLOT "VP_CWORD63_CWORD84_SDslot"
#define VP_CWORD63_BTANT_TYPE "VP_CWORD63_BTANT_TYPE"
#define VP_CWORD63_GVIF_FROM_CWORD80 "VP_CWORD63_GVIF_FROM_CWORD80"
#define VP_CWORD63_MIC_MFR "VP_CWORD63_MIC_MFR"
#define VP_CWORD63_SER_SERIAL "VP_CWORD63_SER_serial"
#define VP_CWORD63_SER_CWORD49_ID "VP_CWORD63_SER_CWORD49_ID"
#define VP_CWORD63_SER_HELPNET_ID "VP_CWORD63_SER_HELPNET_ID"
#define VP_CWORD63_SER_MAP_UPDATE_ID "VP_CWORD63_SER_map_update_ID"
#define VP_CWORD63_SER_SHIP_DATE "VP_CWORD63_SER_ship_date"
#define VP_CWORD63_SER_FACTORY_ID "VP_CWORD63_SER_FACTORY_ID"
#define VP_CWORD31_VERSION "VP_CWORD31_version"
#define VP_CWORD31_CLASS_TYPE_OF_LAN "VP_CWORD31_class_type_of_LAN"
#define VP_CWORD31_REVERSE "VP_CWORD31_reverse"
#define VP_CWORD31_PKB "VP_CWORD31_PKB"
#define VP_CWORD31_ADIM_CWORD62 "VP_CWORD31_ADIM_CWORD62"
#define VP_CWORD31_TELEMATICS_FUNCTION "VP_CWORD31_telematics_function"
#define VP_CWORD31_OPENING_PICTURE "VP_CWORD31_opening_picture"
#define VP_CWORD31_THEME_COLOR "VP_CWORD31_theme_color"
#define VP_CWORD31_AC_ONSCREEN "VP_CWORD31_AC_onscreen"
#define VP_CWORD31_AC_SCREEN_OPERATION "VP_CWORD31_AC_screen_operation"
#define VP_CWORD31_AC_SCREEN_HOME "VP_CWORD31_AC_screen_home"
#define VP_CWORD31_AC_SWITCH_FRONT_TEMP "VP_CWORD31_AC_switch_front_temp"
#define VP_CWORD31_AC_TEMP_DISP_OUTSIDE "VP_CWORD31_AC_temp_disp_outside"
#define VP_CWORD31_AC_TEMP_DISP_SETTING "VP_CWORD31_AC_temp_disp_setting"
#define VP_CWORD31_AC_SWITCH_CONCIERGE "VP_CWORD31_AC_switch_concierge"
#define VP_CWORD31_AC_SWITCH_S_FLOW "VP_CWORD31_AC_switch_S_flow"
#define VP_CWORD31_CWORD11 "VP_CWORD31_CWORD11"
#define VP_CWORD31_CWORD11_HOME "VP_CWORD31_CWORD11_home"
#define
    VP_CWORD31_CWORD11_DRIVE_WHEEL "VP_CWORD31_CWORD11_drive_wheel"

```

```

#define VP_CWORD31_CWORD11_DRIVE_TYPE "VP_CWORD31__CWORD11_drive_type"
#define
  VP_CWORD31_CWORD11_BATTERY_POSITION "VP_CWORD31__CWORD11_battery_po
  sition"
#define
  VP_CWORD31_CWORD11_BATTERY_DIRECTION "VP_CWORD31__CWORD11_battery_d
  irection"
#define
  VP_CWORD31_CWORD11_MOTOR_POSITION "VP_CWORD31__CWORD11_motor_posi
  tion"
#define VP_CWORD31_CWORD9_TRIP "VP_CWORD31__CWORD9_trip"
#define VP_CWORD31_CWORD9_HOME_TRIP "VP_CWORD31__CWORD9_home_trip"
#define VP_CWORD31_CWORD9_FUEL "VP_CWORD31__CWORD9_fuel"
#define VP_CWORD31_CWORD9_HOME_FUEL "VP_CWORD31__CWORD9_home_fuel"
#define
  VP_CWORD31_FC_ECO_SAFETY_POINT_SERVICE "VP_CWORD31_FC_eco_safety_point_ser
  vice"
#define VP_CWORD31_FC_SCALE_GAS "VP_CWORD31_FC_scale_GAS"
#define VP_CWORD31_FC_SCALE_HV "VP_CWORD31_FC_scale_HV"
#define VP_CWORD31_FC_ENERGY_MARK "VP_CWORD31_FC_energy_mark"
#define VP_CWORD31_CWORD9_TYPE "VP_CWORD31__CWORD9_type"
#define VP_CWORD31_FC_MET_CWORD90_04 "VP_CWORD31_FC_MET_CWORD90_04"
#define VP_CWORD31_DISPLAY_SETTING_SCREEN "VP_CWORD31__display_setting_screen"
#define VP_CWORD31_CWORD49_BRAND "VP_CWORD31__CWORD49_brand"
#define VP_CWORD31_VR_HVAC_CONTROL "VP_CWORD31_VR_HVAC_control"
#define VP_CWORD31_CLOCK_TYPE "VP_CWORD31__clock_type"
#define VP_CWORD31_START_UP_BGM "VP_CWORD31__start_up_BGM"
#define VP_CWORD31_START_UP_BGM_TYPE "VP_CWORD31__start_up_BGM_type"
#define VP_CWORD31_DTV_ANTENNA_FL "VP_CWORD31__DTV_antenna_FL"
#define VP_CWORD31_DTV_ANTENNA_FR "VP_CWORD31__DTV_antenna_FR"
#define VP_CWORD31_DTV_ANTENNA_RL "VP_CWORD31__DTV_antenna_RL"
#define VP_CWORD31_DTV_ANTENNA_RR "VP_CWORD31__DTV_antenna_RR"
#define VP_CWORD31_HF_VOICE_QUALITY_TYPE "VP_CWORD31__HF_voice_quality_type"
#define VP_CWORD31_TUNE_KNOB_PUSH "VP_CWORD31__tune_knob_push"
#define VP_CWORD31_SCREEN_SWITCH_SEEK "VP_CWORD31__screen_switch_seek"
#define VP_CWORD31_DAB_ANTENNA "VP_CWORD31__DAB_antenna"
#define VP_CWORD31_HARD_SWITCH_BEEP "VP_CWORD31__hard_switch_beep"
#define VP_CWORD31_SCREEN_SWITCH_BEEP "VP_CWORD31__screen_switch_beep"
#define VP_CWORD31_STEERING_SWITCH_TYPE "VP_CWORD31__steering_switch_type"
#define VP_CWORD31_STEERING_SWITCH_TEL "VP_CWORD31__steering_switch_tel"
#define VP_CWORD31_CLASS_CIRCUIT_MODE "VP_CWORD31__class_circuit_mode"
#define VP_CWORD31_CWORD67_WARNING "VP_CWORD31__CWORD67_warning"
#define VP_CWORD31_CWORD67_ACCELERATOR "VP_CWORD31__CWORD67_accelerator"
#define VP_CWORD31_PANEL_MICROCOMPUTER "VP_CWORD31__panel_microcomputer"
#define
  VP_CWORD31_CLASS_WATCH_AT_BRAKE_HOLD "VP_CWORD31__class_watch_at_brake_ho
  ld"
#define VP_CWORD31_GPS_ANTENNA "VP_CWORD31__GPS_antenna"

```

```

#define VP_CWORD31_VEHICLE_SIGNAL_ILL "VP_CWORD31_vehicle_signal_ILL"
#define VP_CWORD31_SWITCH_TRIGGER "VP_CWORD31_switch_trigger"
#define VP_CWORD31_SENSITIVITY_LEVEL "VP_CWORD31_sensitivity_level"
#define VP_CWORD31_INVALIDITY_TERM "VP_CWORD31_invalidity_term"
#define VP_CWORD31_THUMBNAI_TYPE "VP_CWORD31_thumbnail_type"
#define VP_CWORD31_BT_DEVICE_NAME "VP_CWORD31_BT_device_name"
#define VP_CWORD31_ECO_DRIVE_MAX_VOLUME "VP_CWORD31_eco_drive_max_volume"
#define VP_CWORD31_EOM_FUNCTION "VP_CWORD31_EOM_function"
#define VP_CWORD31_MICROPHONE_TYPE "VP_CWORD31_Microphone_Type"
#define VP_CWORD31_MICROPHONE_LOCATION "VP_CWORD31_Microphone_Location"
#define VP_CWORD31_WIDE_BAND_FM "VP_CWORD31_Wide_band_FM"
#define VP_CWORD31_MAP_COLOR "VP_CWORD31_map_color"
#define VP_CWORD31_CLOUD_ECO "VP_CWORD31_cloud_eco"
#define VP_CWORD31_EG_TYPE "VP_CWORD31_EG_type"
#define VP_CWORD31_DISPLAY_TYPE "VP_CWORD31_display_type"
#define VP_CWORD31_CLOCK_DISPLAY "VP_CWORD31_clock_display"
#define VP_CWORD31_DESTINATION "VP_CWORD31_destination"
#define VP_CWORD31_CLASS_OPERATION "VP_CWORD31_class_operation"
#define VP_CWORD31_OPENING_INTERVAL "VP_CWORD31_opening_interval"
#define VP_CWORD31_DES_FUNCTION "VP_CWORD31_DES_function"
#define VP_CWORD31_BEEP_SOUND_KEYS "VP_CWORD31_beep_sound_keys"
#define XM_AUDIO ((u_int8_t)1)
#define CWORD76 ((u_int8_t)2)
#define VP_CANRCV_CWORD27 ((u_int8_t)3)
#define WIFI_HOTSPOT ((u_int8_t)4)
#define NA_STD_TRAF_LIMIT ((u_int8_t)5)
#define HD_DATA ((u_int8_t)6)
#define FUNCTION_CHECK_RESULT_OK ((u_int8_t)0)
#define FUNCTION_CHECK_RESULT_NG ((u_int8_t)1)
#define FUNCTION_CHECK_RESULT_UNKNOWN ((u_int8_t)2)
#define FUNCTION_CHECK_RESULT_OUT_OF_NA ((u_int8_t)3)
#define VP_MAX_LENGTH 128
#define CWORD63_AREA "_CWORD63_AREA"
#define CWORD63_GRADE "_CWORD63_GRADE"
#define CWORD63_UNIT "_CWORD63_UNIT"
#define CWORD63_BRAND "_CWORD63_BRAND"
#define CWORD63_HAS_CWORD80 "_CWORD63_HAS_CWORD80_"
#define CWORD63_BOARD "_CWORD63_BOARD"
#define CWORD63_CWORD86 "_CWORD63_CWORD86_"

```

Functions

```

u_int8_t VP_FuncCheck_CanRcv (u_int8_t FunctionName)
void VP_GetEnv (const char *pEnvStr, char *pEnvBuff)

```

Detailed Description



Macro Definition Documentation

#define _CWORD63__CWORD86_ "_CWORD63__CWORD86_"

CWORD86 type : "y" or "-"

#define _CWORD63__AREA "_CWORD63__AREA"

definition of parameter environment variable such as destination destination diff :
"na" or "jp" or "eu" or "ge", others are not decide

#define _CWORD63__BOARD "_CWORD63__BOARD"

board type : "_CWORD71_1s" or "_CWORD71__CWORD90_" or
"_CWORD71__CWORD91_"

#define _CWORD63__BRAND "_CWORD63__BRAND"

brand diff : "_CWORD45_" or "_CWORD31_"

#define _CWORD63__GRADE "_CWORD63__GRADE"

grade diff : "_CWORD96_" or "_CWORD95_" or "I1" or "_CWORD101_"

#define _CWORD63__HAS_CWORD80_ "_CWORD63__HAS_CWORD80_"

CWORD80 is connected or not : "y" or "n"

#define _CWORD63__UNIT "_CWORD63__UNIT"

unit diff: "_CWORD84_" or "_CWORD80_"

#define _CWORD76_ ((u_int8_t)2)

CWORD76

#define FUNCTION_CHECK_RESULT_NG ((u_int8_t)1)

this function is invalid

#define FUNCTION_CHECK_RESULT_OK ((u_int8_t)0)

< *CWORD45* Enform < *CWORD31* Enform definition of return value of
API(VP_FuncCheck_CanRcv), which is used to get status of function implementation

#define FUNCTION_CHECK_RESULT_OUT_OF_NA ((u_int8_t)3)

reference out of north american

#define FUNCTION_CHECK_RESULT_UNKNOWN ((u_int8_t)2)

this function does not exist

#define HD_DATA ((u_int8_t)6)

HD Data (Include traffic)

#define NA_STD_TRAF_LIMIT ((u_int8_t)5)

standard driving rregulation of north america

#define VP_CWORD31__CWORD11_ "VP_CWORD31__CWORD11_"

CWORD31 EF screen

#define

VP_CWORD31__CWORD11_BATTERY_DIRECTION "VP_CWORD31__CWORD11_battery_ direction"

CWORD31 battery direction of EF screen

#define

VP_CWORD31__CWORD11_BATTERY_POSITION "VP_CWORD31__CWORD11_battery_p osition"

CWORD31 battery position of EF screen

#define

VP_CWORD31__CWORD11_DRIVE_TYPE "VP_CWORD31__CWORD11_drive_type"

CWORD31 drive type of EF screen

#define

VP_CWORD31__CWORD11_DRIVE_WHEEL "VP_CWORD31__CWORD11_drive_wheel"

CWORD31 drive wheel of EF screen

#define VP_CWORD31__CWORD11_HOME "VP_CWORD31__CWORD11_home"

CWORD31 EF home screen

#define

VP_CWORD31__CWORD11_MOTOR_POSITION "VP_CWORD31__CWORD11_motor_pos ition"

CWORD31 motor position of EF screen

#define VP_CWORD31__CWORD49_BRAND "VP_CWORD31__CWORD49_brand"

CWORD31 CWORD49 brand

#define

VP_CWORD31__CWORD67_ACCELERATOR "VP_CWORD31__CWORD67_accelerator"

CWORD31 CWORD67 accelerator

#define VP_CWORD31__CWORD67_WARNING "VP_CWORD31__CWORD67_warning"

CWORD31 CWORD67 warning

#define VP_CWORD31__CWORD9_FUEL "VP_CWORD31__CWORD9_fuel"

CWORD31 FC screen fuel

#define VP_CWORD31__CWORD9_HOME_FUEL "VP_CWORD31__CWORD9_home_fuel"

CWORD31 FC home screen fuel

#define VP_CWORD31__CWORD9_HOME_TRIP "VP_CWORD31__CWORD9_home_trip"

CWORD31 FC home screen trip

#define VP_CWORD31__CWORD9_TRIP "VP_CWORD31__CWORD9_trip"

CWORD31 FC screen trip

#define VP_CWORD31__CWORD9_TYPE "VP_CWORD31__CWORD9_type"

CWORD31 FC screen type

#define VP_CWORD31__AC_ONSCREEN "VP_CWORD31__AC_onscreen"

CWORD31 air conditioner on screen

#define VP_CWORD31__AC_SCREEN_HOME "VP_CWORD31__AC_screen_home"

CWORD31 air conditioner home screen

#define VP_CWORD31__AC_SCREEN_OPERATION "VP_CWORD31__AC_screen_operation"

CWORD31 air conditioner screen operation

#define VP_CWORD31__AC_SWITCH_CONCIERGE "VP_CWORD31__AC_switch_concierge"

CWORD31 air conditioner switch concierge

#define

VP_CWORD31__AC_SWITCH_FRONT_TEMP "VP_CWORD31__AC_switch_front_temp"

CWORD31 air conditioner switch front temperature

#define VP_CWORD31_AC_SWITCH_S_FLOW "VP_CWORD31_AC_switch_S_flow"

S flow of *CWORD31* air conditioner switch

#define

VP_CWORD31_AC_TEMP_DISP_OUTSIDE "VP_CWORD31_AC_temp_disp_outside"

CWORD31 air conditioner display of outside temperature

#define VP_CWORD31_AC_TEMP_DISP_SETTING "VP_CWORD31_AC_temp_disp_setting"

CWORD31 air conditioner setting of display

#define VP_CWORD31_ADIM_CWORD62_ "VP_CWORD31_ADIM_CWORD62_"

CWORD31 ADIM signal of *CWORD62*

#define VP_CWORD31_BEEP_SOUND_KEYS "VP_CWORD31_beep_sound_keys"

CWORD31 beep sound keys

#define VP_CWORD31_BT_DEVICE_NAME "VP_CWORD31_BT_device_name"

CWORD31 BT device name

#define VP_CWORD31_CLASS_CIRCUIT_MODE "VP_CWORD31_class_circuit_mode"

CWORD31 class circuit mode

#define VP_CWORD31_CLASS_OPERATION "VP_CWORD31_class_operation"

CWORD31 class operation

#define VP_CWORD31_CLASS_TYPE_OF_LAN "VP_CWORD31_class_type_of_LAN"

CWORD31 class type of LAN

#define

VP_CWORD31_CLASS_WATCH_AT_BRAKE_HOLD "VP_CWORD31_class_watch_at_brake_hold"

CWORD31 class watch at brake hold

#define VP_CWORD31_CLOCK_DISPLAY "VP_CWORD31_clock_display"

CWORD31 clock display

```

#define VP_CWORD31_CLOCK_TYPE "VP_CWORD31_clock_type"

    CWORD31 clock type

#define VP_CWORD31_DAB_ANTENNA "VP_CWORD31_DAB_antenna"

    CWORD31 DAB antenna

#define VP_CWORD31_DES_FUNCTION "VP_CWORD31_DES_function"

    CWORD31 DES function

#define VP_CWORD31_DESTINATION "VP_CWORD31_destination"

    CWORD31 destination

#define
VP_CWORD31_DISPLAY_SETTING_SCREEN "VP_CWORD31_display_setting_screen"

    CWORD31 display setting screen

#define VP_CWORD31_DISPLAY_TYPE "VP_CWORD31_display_type"

    CWORD31 display type

#define VP_CWORD31_DTV_ANTENNA_FL "VP_CWORD31_DTV_antenna_FL"

    CWORD31 font left DTV antenna

#define VP_CWORD31_DTV_ANTENNA_FR "VP_CWORD31_DTV_antenna_FR"

    CWORD31 font right DTV antenna

#define VP_CWORD31_DTV_ANTENNA_RL "VP_CWORD31_DTV_antenna_RL"

    CWORD31 rear left DTV antenna

#define VP_CWORD31_DTV_ANTENNA_RR "VP_CWORD31_DTV_antenna_RR"

    CWORD31 rear right DTV antenna

#define
VP_CWORD31_ECO_DRIVE_MAX_VOLUME "VP_CWORD31_eco_drive_max_volume"

    CWORD31 eco drive max volume

#define VP_CWORD31_EOM_FUNCTION "VP_CWORD31_EOM_function"

    CWORD31 EOM function

```

```

#define
VP_CWORD31_FC_ECO_SAFETY_POINT_SERVICE "VP_CWORD31_FC_eco_safety_point_ser
vice"

    CWORD31 FC eco safety point service

#define VP_CWORD31_FC_ENERGY_MARK "VP_CWORD31_FC_energy_mark"

    CWORD31 FC energy mark

#define VP_CWORD31_FC_MET_CWORD90_04 "VP_CWORD31_FC_MET_CWORD90_04"

    CWORD31 FC MET_CWORD90_04 signal

#define VP_CWORD31_FC_SCALE_GAS "VP_CWORD31_FC_scale_GAS"

    CWORD31 FC GAS scale

#define VP_CWORD31_FC_SCALE_HV "VP_CWORD31_FC_scale_HV"

    CWORD31 FC HV scale

#define VP_CWORD31_GPS_ANTENNA "VP_CWORD31_GPS_antenna"

    CWORD31 GPS antenna

#define VP_CWORD31_HARD_SWITCH_BEEP "VP_CWORD31_hard_switch_beep"

    CWORD31 hard switch beep

#define
VP_CWORD31_HF_VOICE_QUALITY_TYPE "VP_CWORD31_HF_voice_quality_type"

    CWORD31 HF voice quality type

#define VP_CWORD31_INVALIDITY_TERM "VP_CWORD31_invalidity_term"

    CWORD31 invalidity term

#define
VP_CWORD31_MICROPHONE_LOCATION "VP_CWORD31_Microphone_Location"

    CWORD31 Microphone Location

#define VP_CWORD31_MICROPHONE_TYPE "VP_CWORD31_Microphone_Type"

    CWORD31 Microphone Type

#define VP_CWORD31_OPENING_INTERVAL "VP_CWORD31_opening_interval"

```

CWORD31 opening interval

#define VP_CWORD31_OPENING_PICTURE "VP_CWORD31_opening_picture"

CWORD31 opening picture

#define

VP_CWORD31_PANEL_MICROCOMPUTER "VP_CWORD31_panel_microcomputer"

CWORD31 panel microcomputer

#define VP_CWORD31_PKB "VP_CWORD31_PKB"

CWORD31 PKB signal

#define VP_CWORD31_REVERSE "VP_CWORD31_reverse"

CWORD31 reverse signal

#define VP_CWORD31_SCREEN_SWITCH_BEEP "VP_CWORD31_screen_switch_beep"

CWORD31 screen switch beep

#define VP_CWORD31_SCREEN_SWITCH_SEEK "VP_CWORD31_screen_switch_seek"

CWORD31 screen switch seek

#define VP_CWORD31_SENSITIVITY_LEVEL "VP_CWORD31_sensitivity_level"

CWORD31 sensitivity level

#define VP_CWORD31_START_UP_BGM "VP_CWORD31_start_up_BGM"

CWORD31 start up BGM

#define VP_CWORD31_START_UP_BGM_TYPE "VP_CWORD31_start_up_BGM_type"

CWORD31 start up BGM type

#define VP_CWORD31_STEERING_SWITCH_TEL "VP_CWORD31_steering_switch_tel"

CWORD31 TEL switch of steering

#define VP_CWORD31_STEERING_SWITCH_TYPE "VP_CWORD31_steering_switch_type"

CWORD31 steering switch type

#define VP_CWORD31_SWITCH_TRIGGER "VP_CWORD31_switch_trigger"

CWORD31 switch trigger

#define VP_CWORD31_TELEMATICS_FUNCTION "VP_CWORD31_telematics_function"

CWORD31 telematics function

#define VP_CWORD31_THEME_COLOR "VP_CWORD31_theme_color"

CWORD31 theme color

#define VP_CWORD31_THUMBNAIL_TYPE "VP_CWORD31_thumbnail_type"

CWORD31 thumbnail type

#define VP_CWORD31_TUNE_KNOB_PUSH "VP_CWORD31_tune_knob_push"

CWORD31 tune knob push

#define VP_CWORD31_VEHICLE_SIGNAL_ILL "VP_CWORD31_vehicle_signal_ILL"

CWORD31 illumination vehicle signal

#define VP_CWORD31_VERSION "VP_CWORD31_version"

CWORD31 version

#define VP_CWORD31_VR_HVAC_CONTROL "VP_CWORD31_VR_HVAC_control"

CWORD31 VR HVAC control

#define VP_CWORD31_WIDE_BAND_FM "VP_CWORD31_Wide_band_FM"

CWORD31 wide band FM

#define

VP_CWORD55_CWORD66_SVR_PRODUCT_BROWSER_URL "VP_CWORD55_CWORD66_SVR_PRODUCT_BROWSER_URL"

CWORD55 Browser URL of Product for *CWORD66* .

#define

VP_CWORD55_CWORD66_SVR_PRODUCT_DOWNLOAD_HOST "VP_CWORD55_CWORD66_SVR_PRODUCT_DOWNLOAD_HOST"

CWORD55 Download hostname of Product for *CWORD66* .

#define

VP_CWORD55_CWORD66_SVR_PRODUCT_DOWNLOAD_URL "VP_CWORD55_CWORD66_SVR_PRODUCT_DOWNLOAD_URL"

CWORD55 Download URL of Product for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_PRODUCT_UPLOAD_HOST "VP_CWORD55__CWORD66__SVR_PRODUCT_UPLOAD_HOST"

CWORD55 Upload hostname of Product for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_PRODUCT_UPLOAD_URL "VP_CWORD55__CWORD66__SVR_PRODUCT_UPLOAD_URL"

CWORD55 Upload URL of Product for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TEST_BROWSER_URL "VP_CWORD55__CWORD66__SVR_TEST_BROWSER_URL"

CWORD55 Browser URL of Test for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TEST_DOWNLOAD_HOST "VP_CWORD55__CWORD66__SVR_TEST_DOWNLOAD_HOST"

CWORD55 Download hostname of Test for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TEST_DOWNLOAD_URL "VP_CWORD55__CWORD66__SVR_TEST_DOWNLOAD_URL"

CWORD55 Download URL of Test for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TEST_UPLOAD_HOST "VP_CWORD55__CWORD66__SVR_TEST_UPLOAD_HOST"

CWORD55 Upload hostname of Test for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TEST_UPLOAD_URL "VP_CWORD55__CWORD66__SVR_TEST_UPLOAD_URL"

CWORD55 Upload URL of Test for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TRY_BROWSER_URL "VP_CWORD55__CWORD66__SVR_TRY_BROWSER_URL"

CWORD55 Browser URL of Try for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TRY_DOWNLOAD_HOST "VP_CWORD55__CWORD66__SVR_TRY_DOWNLOAD_HOST"

CWORD55 Download hostname of Try for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TRY_DOWNLOAD_URL "VP_CWORD55__CWORD66__SVR_TRY_DOWNLOAD_URL"

CWORD55 Download URL of Try for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TRY_UPLOAD_HOST "VP_CWORD55__CWORD66__SVR_TRY_UPLOAD_HOST"

CWORD55 Upload hostname of Try for *CWORD66* .

#define

VP_CWORD55__CWORD66_SVR_TRY_UPLOAD_URL "VP_CWORD55__CWORD66__SVR_TRY_UPLOAD_URL"

CWORD55 Upload URL of Try for *CWORD66* .

#define

VP_CWORD55__REPRO_SVR_PRODUCT_URL "VP_CWORD55__REPRO_SVR_PRODUCT_URL"

CWORD55 URL of Product for Repro.

#define VP_CWORD55__REPRO_SVR_TEST_URL "VP_CWORD55__REPRO_SVR_TEST_URL"

CWORD55 URL of Test for Repro.

#define VP_CWORD55__REPRO_SVR_TRY_URL "VP_CWORD55__REPRO_SVR_TRY_URL"

CWORD55 URL of Try for Repro.

#define VP_CWORD63__CWORD105_ "VP_CWORD63__CWORD105_"

CWORD63 *CWORD105*

#define VP_CWORD63__CWORD17_ "VP_CWORD63__CWORD17_"

CWORD63 *CWORD17*

#define VP_CWORD63__CWORD32_DAB "VP_CWORD63__CWORD32_DAB"

CWORD63 *CWORD32* *DAB*

```

#define
VP_CWORD63__CWORD33_LOG_ENC_KEYTYPE "VP_CWORD63__CWORD33_LOG_ENC_KE
YTYPE"

    CWORD63 _CWORD33_log ENC key type

#define VP_CWORD63__CWORD63_ID "VP_CWORD63__CWORD63_ID"

    CWORD63 ID

#define VP_CWORD63__CWORD6_ "VP_CWORD63__CWORD6_"

    CWORD63 hybrid VR

#define VP_CWORD63__AMP_TYPE "VP_CWORD63__AMP_type"

    CWORD63 amp type

#define VP_CWORD63__AREA "VP_CWORD63__area"

    CWORD63 area

#define VP_CWORD63__BT_2HFP "VP_CWORD63__BT_2HFP"

    CWORD63 BT HFP

#define VP_CWORD63__BT_ADD_DEVICE_SW "VP_CWORD63__BT_ADD_DEVICE_SW"

    CWORD63 BT add device switch

#define VP_CWORD63__CACHERADIO "VP_CWORD63__CacheRadio"

    CWORD63 cache radio

#define VP_CWORD63__CERTIFICATION_SOFT "VP_CWORD63__CERTIFICATION_SOFT"

    CWORD63 certification soft

#define VP_CWORD63__CONNECTION_SERVER "VP_CWORD63__CONNECTION_SERVER"

    CWORD63 connection server

#define VP_CWORD63__CONTACT "VP_CWORD63__CONTACT"

    CWORD63 contact

#define VP_CWORD63__DAB_SUB_ANTENNA "VP_CWORD63__DAB_sub_antenna"

    CWORD63 DAB sub antenna

```

#define VP_CWORD63_DAB_TUNER_NUM "VP_CWORD63_DAB_tuner_num"

CWORD63 DAB tuner number

#define VP_CWORD63_DECK "VP_CWORD63_DECK"

CWORD63 DECK

#define VP_CWORD63_DEFAULT_MODE "VP_CWORD63_default_mode"

CWORD63 display mode

#define VP_CWORD63_DES_VOL_1 "VP_CWORD63_DES_VOL_1"

CWORD63 destination vol 1

#define VP_CWORD63_DES_VOL_2 "VP_CWORD63_DES_VOL_2"

CWORD63 destination vol 2

#define VP_CWORD63_DES_VOL_3 "VP_CWORD63_DES_VOL_3"

CWORD63 destination vol 3

#define VP_CWORD63_DES_VOL_4 "VP_CWORD63_DES_VOL_4"

CWORD63 destination vol 4

#define VP_CWORD63_DES_VOL_5 "VP_CWORD63_DES_VOL_5"

CWORD63 destination vol 5

#define VP_CWORD63_DES_VOL_6 "VP_CWORD63_DES_VOL_6"

CWORD63 destination vol 6

#define VP_CWORD63_DES_VOL_7 "VP_CWORD63_DES_VOL_7"

CWORD63 destination vol 7

#define VP_CWORD63_DES_VOL_8 "VP_CWORD63_DES_VOL_8"

CWORD63 destination vol 8

#define VP_CWORD63_DEST_C_CODE "VP_CWORD63_dest_c_code"

CWORD63 destination country code

```

#define
VP_CWORD63_DISCONNECT_CWORD80_TIMEOUT "VP_CWORD63_DISCONNECT_CW
ORD80_TIMEOUT"

    CWORD63 CWORD79 disconnect timeout

#define VP_CWORD63_DISPLAY "VP_CWORD63_DISPLAY"

    CWORD63 display

#define VP_CWORD63_GRADE "VP_CWORD63_grade"

    CWORD63 grade

#define
VP_CWORD63_HANDWRITEMODE_MOVERANGE "VP_CWORD63_Handwritemode_mov
erange"

    CWORD63 move range of handwrite mode

#define
VP_CWORD63_HAS_CWORD80_DEFAULT "VP_CWORD63_HAS_CWORD80_DEFAULT"

    CWORD63 default status is has CWORD80

#define VP_CWORD63_HAS_GPS "VP_CWORD63_HAS_GPS"

    CWORD63 has GPS

#define
VP_CWORD63_INFOMATION_MANAGEMENT "VP_CWORD63_INFOMATION_MANAGE
MENT"

    CWORD63 information management

#define VP_CWORD63_JPTRAFFICINFO_SHOW "VP_CWORD63_JPTrafficInfo_show"

    CWORD63 JP traffic information show

#define VP_CWORD63_MAKER_ID "VP_CWORD63_maker_ID"

    CWORD63 maker ID

#define VP_CWORD63_MODEL_ID "VP_CWORD63_model_ID"

    CWORD63 model ID

#define VP_CWORD63_OP_COOPERATION "VP_CWORD63_OP_cooperation"

```

CWORD63 op cooperation

#define VP_CWORD63_PHASE "VP_CWORD63_PHASE"

CWORD63 phase

#define VP_CWORD63_RADIO_ANTENNA_NUM "VP_CWORD63_radio_antenna_num"

CWORD63 radio antenna number

#define VP_CWORD63_RADIO_TUNER_NUM "VP_CWORD63_radio_tuner_num"

CWORD63 radio tuner number

#define VP_CWORD63_SEARCH "VP_CWORD63_SEARCH"

CWORD63 search

#define VP_CWORD63_SER_CWORD49_ID "VP_CWORD63_SER_CWORD49_ID"

Environment variable *_CWORD49_ID*

#define VP_CWORD63_SER_FACTORY_ID "VP_CWORD63_SER_FACTORY_ID"

Environment variable *FACTORY_ID*

#define VP_CWORD63_SER_HELPNET_ID "VP_CWORD63_SER_HELPNET_ID"

Environment variable *HELPNET_ID*

#define VP_CWORD63_SER_MAP_UPDATE_ID "VP_CWORD63_SER_map_update_ID"

Environment variable *map_update_ID*

#define VP_CWORD63_SER_SERIAL "VP_CWORD63_SER_serial"

Environment variable *serial*

#define VP_CWORD63_SER_SHIP_DATE "VP_CWORD63_SER_ship_date"

Environment variable *ship_date*

#define VP_CWORD63_STEERING_TYPE "VP_CWORD63_STEERING_TYPE"

CWORD63 steering type

#define VP_CWORD63_SUBSCREEN_APPS_SW "VP_CWORD63_SUBSCREEN_APPS_SW"

CWORD63 sub screen apps show

#define VP_CWORD63_TOUCHPAD_HOME_SW "VP_CWORD63_TOUCHPAD_HOME_SW"

CWORD63 touchpad home show

#define VP_CWORD63_TOUCHPANEL "VP_CWORD63_TOUCHPANEL"

CWORD63 touch panel

#define VP_CWORD63_VERSION "VP_CWORD63_version"

CWORD63 version

#define

VP_CWORD63_VOICE_RECOGNITION_TUTORIAL "VP_CWORD63_VOICE_RECOGNITION_TUTORIAL"

CWORD63 voice recognition tutorial

#define VP_CWORD63_VOICE_TAGS "VP_CWORD63_VOICE_TAGS"

CWORD63 voice tags

#define VP_CWORD63_WIFI_AP_ENABLE "VP_CWORD63_WIFI_AP_ENABLE"

CWORD63 WIFI AP enable

#define VP_CWORD63_WIFI_DIRECT_ENABLE "VP_CWORD63_WIFI_DIRECT_ENABLE"

CWORD63 WIFI DIRECT enable

#define VP_CWORD63_WIFI_STA_ENABLE "VP_CWORD63_WIFI_STA_ENABLE"

CWORD63 WIFI STA enable

#define VP_CANRCV_CWORD27_ ((u_int8_t)3)

WiFi HotSpot standard driving rregulation of north america HD Data (Include traffic)_CWORD27_

#define VP_MAX_LENGTH 128

the max length of vehicle parameter environment viarable

#define WIFI_HOTSPOT ((u_int8_t)4)

WiFi HotSpot

#define XM_AUDIO ((u_int8_t)1)

Function Documentation

u_int8_t VP_FuncCheck_CanRcv (u_int8_t *FunctionName*)

Brief

VP_FuncCheck_CanRcv

Parameters:

in	<i>FunctionName</i>	function name
----	---------------------	---------------

Return values:

<i>FUNCTION_CHECK_RESULT_OK</i>	This function is available
<i>FUNCTION_CHECK_RESULT_NG</i>	This function is unavailable
<i>FUNCTION_CHECK_RESULT_UNKNOWN</i>	This function does not exist

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

void VP_GetEnv (const char * *pEnvStr*, char * *pEnvBuff*)

definition of *CWORD31* vehicle parameter environment variable definition of *CWORD63* vehicle parameter environment variable definition of *CWORD63* (Serial) vehicle parameter environment variable

Brief

VP_GetEnv

Parameters:

in	<i>pEnvStr</i>	The pointer of environment variable name string
out	<i>pEnvBuff</i>	The head address of saving environment variable value

Return values:

<i>None</i>	
-------------	--

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

Event_libraryEV_ERR [EV_create_flag](#) (EV_ID flag_id)EV_ERR [EV_create_flag64](#) (EV_ID flag_id)EV_ERR [EV_create_queue](#) (EV_ID queue_id, UINT8 length, UINT16 max_bytes,
EV_Message_Queue_Type type)EV_ERR [EV_create_flag_auto_id](#) (EV_ID *flag_id)EV_ERR [EV_create_flag64_auto_id](#) (EV_ID *flag_id)EV_ERR [EV_create_queue_auto_id](#) (EV_ID *queue_id, UINT8 length, UINT16 max_bytes,
EV_Message_Queue_Type type)EV_ERR [EV_destroy_flag](#) (EV_ID queue_id)EV_ERR [EV_destroy_queue](#) (EV_ID queue_id)EV_ERR [EV_set_flag](#) (EV_ID flag_id, UINT32 bits)EV_ERR [EV_set_flag64](#) (EV_ID flag_id, UINT64 bits)EV_ERR [EV_send_message](#) (EV_ID queue_id, UINT16 bytes, const void *message, UINT32
senderInfo)EV_ERR [EV_get_next_event](#) (EV_Event *ev)EV_ERR [EV_peek_next_event](#) (EV_Event *ev)EV_ERR [EV_get_flag](#) (EV_ID flag_id, EV_Flag *flag)EV_ERR [EV_wait_flag](#) (EV_ID flag_id, EV_Flag *flag)EV_ERR [EV_peek_flag](#) (EV_ID flag_id, EV_Flag *flag)EV_ERR [EV_get_flag64](#) (EV_ID flag_id, EV_Flag64 *flag)EV_ERR [EV_wait_flag64](#) (EV_ID flag_id, EV_Flag64 *flag)EV_ERR [EV_peek_flag64](#) (EV_ID flag_id, EV_Flag64 *flag)EV_ERR [EV_get_message](#) (EV_ID queue_id, EV_Message *message)EV_ERR [EV_wait_message](#) (EV_ID queue_id, EV_Message *message)EV_ERR [EV_peek_message](#) (EV_ID queue_id, EV_Message *message)EV_ERR [EV_find_message_by_sender](#) (EV_ID queue_id, UINT32 senderInfo, EV_Message *message)EV_ERR [EV_find_message_by_content](#) (EV_ID queue_id, UINT16 length, const void *compare_bytes,
EV_Message *message)EV_ERR [EV_get_flag_fd](#) (EV_ID flag_id, int *fd)EV_ERR [EV_get_queue_fd](#) (EV_ID queue_id, int *fd)EV_ID [EV_moduleID_to_flagID](#) (UINT16 m_id)EV_ID [EV_moduleID_to_flag64ID](#) (UINT16 m_id)EV_ID [EV_moduleID_to_queueID](#) (UINT16 m_id)EV_ERR [EV_create_flag_by_mID](#) (UINT16 m_id)

EV_ERR [EV_create_flag64_by_mID](#) (UINT16 m_id)
 EV_ERR [EV_create_queue_by_mID](#) (UINT16 m_id, UINT8 length, UINT16 max_bytes,
 EV_Message_Queue_Type type)
 EV_ERR [EV_destroy_flag_by_mID](#) (UINT16 m_id)
 EV_ERR [EV_destroy_queue_by_mID](#) (UINT16 m_id)
 EV_ERR [EV_set_flag_by_mID](#) (UINT16 m_id, UINT32 bits)
 EV_ERR [EV_set_flag64_by_mID](#) (UINT16 m_id, UINT64 bits)
 EV_ERR [EV_send_message_by_mID](#) (UINT16 m_id, UINT16 bytes, const void *message, UINT32
 senderInfo)

Detailed Description

Function Documentation

EV_ERR EV_create_flag (EV_ID *flag_id*)

Brief

Create flag for event flag by the flag_id.

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID of the flag event
----	----------------	-----------------------------------

EV_ID type parameter
 1 typedef UINT32 EV_ID

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid flag_id
<i>EV_ERR_Thread_Over</i>	The MAX of event thread.
<i>EV_ERR_Exist</i>	Duplication Error
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

There is no prerequisite

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

evk_open, evk_close, evk_ioctl, evk_create_flag, evk_set_poll

EV_ERR EV_create_flag64 (EV_ID *flag_id*)

Brief

Create flag for event flag by the flag_id.

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID of the flag event
----	----------------	-----------------------------------

EV_ID type parameter
1 typedef UINT32 EV_ID

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid flag_id
<i>EV_ERR_Thread_Over</i>	The MAX of event thread.
<i>EV_ERR_Exist</i>	Duplication Error
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

There is no prerequisite

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

evk_open, evk_close, evk_ioctl, evk_create_flag, evk_set_poll

EV_ERR EV_create_flag64_auto_id (EV_ID * *flag_id*)

Brief

Automatically assign flag ID, and creates flag with flag ID.

Parameters:

out	<i>flag_id</i>	EV_ID* - flag_id assigned automatically
-----	----------------	---

EV_ID type parameter
1 typedef UINT32 EV_ID

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Thread_Over</i>	The MAX of event thread
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

There is no prerequisite

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

[EV_create_flag](#), evk_open, evk_close, evk_ioctl, evk_create_flag, evk_alloc_flag_id, evk_set_poll

EV_ERR EV_create_flag64_by_mID (UINT16 *m_id*)**Brief create 64bit flag(moduleID specified)****Parameters:**

in	<i>m_id</i>	moduleID
----	-------------	----------

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid flag_id
<i>EV_ERR_Thread_Over</i>	The MAX of event thread.
<i>EV_ERR_Exist</i>	Duplication Error
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

There is no prerequisite

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

EV_ERR EV_create_flag_auto_id (EV_ID * *flag_id*)

Brief

Automatically assign flag ID, and creates flag with flag ID.

Parameters:

out	<i>flag_id</i>	EV_ID* - <i>flag_id</i> assigned automatically
-----	----------------	--

EV_ID type parameter
1 typedef UINT32 EV_ID

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Thread_Over</i>	The MAX of event thread
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

There is no prerequisite

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

[EV_create_flag](#), [evk_open](#), [evk_close](#), [evk_ioctl](#), [evk_create_flag](#), [evk_alloc_flag_id](#),
[evk_set_poll](#)

EV_ERR EV_create_flag_by_mID (UINT16 *m_id*)

Brief create flag(moduleID specified)

Parameters:

in	<i>m_id</i>	moduleID
----	-------------	----------

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>flag_id</i>
<i>EV_ERR_Thread_Over</i>	The MAX of event thread.
<i>EV_ERR_Exist</i>	Duplication Error
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

There is no prerequisite

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

EV_ERR EV_create_queue (EV_ID *queue_id*, UINT8 *length*, UINT16 *max_bytes*, EV_Message_Queue_Type *type*)

Brief

Create message queue with the *queue_id*.

Parameters:

in	<i>queue_id</i>	EV_ID - <i>queue_id</i> of message event
----	-----------------	--

EV_ID type parameter

1 typedef UINT32 EV_ID

Parameters:

in	<i>length</i>	UINT8 - length of message queueMaximum accumulated number of the messages
in	<i>max_bytes</i>	UINT16 - the maximum number of bytes per 1 message(MAX 2048)
in	<i>type</i>	EV_Message_Queue_Type - classification of the processing when the queue received more event 0 in the queue full state.

EV_Message_Queue_Type

```

1 typedef UINT8 EV_Message_Queue_Type;
2 - EV_MESSAGE_QUEUE_TYPE_BUSY = *** // Return error(EV_ERR_Busy) to origin of message transmission
3 - EV_MESSAGE_QUEUE_TYPE_FIFO = *** // Delete the top message of the queue, and shift message
4 // (Delete the oldest message)
5 - EV_MESSAGE_QUEUE_TYPE_REPLACE = *** // Overwrite the last message of the queue
6 // (Overwrite the newest message)

```

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>queue_id</i>
<i>EV_ERR_Thread_Over</i>	The MAX of event thread.
<i>EV_ERR_Exist</i>	Duplication Error

<i>EV_ERR_Fatal</i>	Abnormal error
---------------------	----------------

Prerequisite

There is no prerequisite

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

evk_open, evk_close, evk_ioctl, evk_create_message_queue, evk_set_poll

EV_ERR EV_create_queue_auto_id (EV_ID * *queue_id*, UINT8 *length*, UINT16 *max_bytes*, EV_Message_Queue_Type *type*)

Brief

Automatically assign queue ID, and creates message queue with queue ID. It stores *queue_id* assigned automatically in argument.

Parameters:

out	<i>queue_id</i>	EV_ID* - <i>queue_id</i> assigned automatically
-----	-----------------	---

EV_ID type parameter
 1 typedef UINT32 EV_ID

Parameters:

in	<i>length</i>	UINT8 - length of message queueMaximum accumulated number of the messages
in	<i>max_bytes</i>	UINT16 - the maximum number of bytes per 1 message(MAX 2048)
in	<i>type</i>	EV_Message_Queue_Type - classification of the processing when the queue received more event in the queue full state.

EV_Message_Queue_Type

```

1 typedef UINT8 EV_Message_Queue_Type;
2 - EV_MESSAGE_QUEUE_TYPE_BUSY = *** // Return error(EV_ERR_Busy) to origin of message transmission
3 - EV_MESSAGE_QUEUE_TYPE_FIFO = *** // Delete the top message of the queue, and shift message
4 // (Delete the oldest message)
5 - EV_MESSAGE_QUEUE_TYPE_REPLACE = *** // Overwrite the last message of the queue
6 // (Overwrite the newest message)

```

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Thread_Over</i>	The MAX of event thread
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

There is no prerequisite

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

[EV_create_queue](#), evk_open, evk_close, evk_ioctl, evk_create_message_queue, evk_alloc_queueID, evk_set_poll

EV_ERR EV_create_queue_by_mID (UINT16 *m_id*, UINT8 *length*, UINT16 *max_bytes*, EV_Message_Queue_Type *type*)

Brief create queue(moduleID specified)**Parameters:**

in	<i>m_id</i>	moduleID
in	<i>length</i>	length of message queue
in	<i>max_bytes</i>	one max bytes for message
in	<i>type</i>	treat type of receiving events even though the queue is full

```

ev_message_queue_type struct
1 enum ev_message_queue_type {
2 EV_MESSAGE_QUEUE_TYPE_BUSY,
3 EV_MESSAGE_QUEUE_TYPE_FIFO,
4 EV_MESSAGE_QUEUE_TYPE_REPLACE,
5 };

```

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_ID</i>	set en invalid flag ID
<i>EV_ERR_Thread_Over</i>	The MAX of event thread.
<i>EV_ERR_Exist</i>	a same flag ID is existing
<i>EV_ERR_Fatal</i>	Fatal error

Prerequisite

There is no prerequisite

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

EV_ERR EV_destroy_flag (EV_ID *queue_id*)**Brief**

Deletes flag with ID specified by the argument.

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID of the flag event
----	----------------	-----------------------------------

EV_ID type parameter
1 typedef UINT32 EV_ID

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>queue_id</i>
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_flag\(\)](#), [EV_create_flag_auto_id\(\)](#) and flag has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

[evk_close](#), [evk_ioctl](#), [evk_destroy_queue](#)

EV_ERR EV_destroy_flag_by_mID (UINT16 *m_id*)**Brief delete flag(moduleID specified)****Parameters:**

in	<i>m_id</i>	moduleID
----	-------------	----------

length [in]length of message queue

max_bytes [in]one max bytes for message

type [in]treat type of receiving events even though the queue is full

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_ID</i>	set en invalid flag ID
<i>EV_ERR_Fatal</i>	Fatal error

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ERR EV_destroy_queue (EV_ID *queue_id*)

Brief

Delete queue for message event by the *queue_id*.

Parameters:

in	<i>queue_id</i>	EV_ID - <i>queue_id</i> of the message event
----	-----------------	--

EV_ID type parameter
1 typedef UINT32 EV_ID

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>queue_id</i>
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_queue\(\)](#), [EV_create_queue_auto_id\(\)](#) and message queue has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:[EV_destroy_flag](#), evk_close, evk_ioctl, evk_destroy_queue**EV_ERR EV_destroy_queue_by_mID (UINT16 *m_id*)****Brief delete queue(moduleID specified)****Parameters:**

in	<i>m_id</i>	moduleID
----	-------------	----------

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid queue_id
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

none

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

EV_ERR EV_find_message_by_content (EV_ID *queue_id*, UINT16 *length*, const void * *compare_bytes*, EV_Message * *message*)**Brief get message event(search with contents)****Parameters:**

in	<i>queue_id</i>	ID of message event queue
in	<i>bytes</i>	size of compare bytes
in	<i>compare_bytes</i>	bytes that used to compare the contents
out	<i>message</i>	message event struct

EV_ID type parameter

```
1 typedef UINT32 EV_ID
```

EV_Message struct

```
1 #define EV_MAX_MESSAGE_LENGTH 2048
2 typedef struct {
3   EV_ID queueID;
4   UINT32 senderInfo;
5   UINT32 length;
6   UINT32 dummy;
7   UINT8 message[EV_MAX_MESSAGE_LENGTH];
8 } EV_Message;
```

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_ID</i>	The specified queue ID is not existing, or it is not created in current process.
<i>EV_ERR_Fatal</i>	Fatal error

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ERR EV_find_message_by_sender (EV_ID *queue_id*, UINT32 *senderInfo*, EV_Message **message*)

Parameters:

in	<i>queue_id</i>	ID of message event queue
in	<i>senderInfo</i>	sender information
out	<i>message</i>	message event struct

EV_ID type parameter

```
1 typedef UINT32 EV_ID
```

EV_Message struct

```

1 #define EV_MAX_MESSAGE_LENGTH 2048
2 typedef struct {
3   EV_ID queueID;
4   UINT32 senderInfo;
5   UINT32 length;
6   UINT32 dummy;
7   UINT8 message[EV_MAX_MESSAGE_LENGTH];
8 } EV_Message;

```

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_ID</i>	The specified queue ID is not existing, or it is not created in current process.
<i>EV_ERR_Fatal</i>	Fatal error

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ERR EV_get_flag (EV_ID *flag_id*, EV_Flag * *flag*)

Brief

Get event flag(non-block)

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID
----	----------------	-----------------

EV_ID type parameter
1 typedef UINT32 EV_ID

Parameters:

out	<i>flag</i>	EV_Flag * - event flag structure
-----	-------------	----------------------------------

EV_Flag structure
1 typedef struct {
2 EV_ID flagID; // Flag ID
3 UINT32 bits; // event flag bit pattern
4 } EV_Flag;

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>flag_id</i>
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_flag\(\)](#), [EV_create_flag_auto_id\(\)](#) and *flag* has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

[evk_ioctl](#), [evk_get_event](#)

EV_ERR EV_get_flag64 (EV_ID *flag_id*, EV_Flag64 * *flag*)**Brief**

Get event flag(non-block)

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID
----	----------------	-----------------

EV_ID type parameter

1 typedef UINT32 EV_ID

Parameters:

out	<i>flag</i>	EV_Flag * - event flag structure
-----	-------------	----------------------------------

EV_Flag structure

```

1 typedef unsigned int UINT32;
2 typedef UINT32 EV_ID
3
4 typedef struct {
5   EV_ID flagID; // Flag ID
6   UINT32 bits; // event flag bit pattern
7 } EV_Flag;

```

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>flag_id</i>
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_flag\(\)](#), [EV_create_flag_auto_id\(\)](#) and flag has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

evk_ioctl, evk_get_event

EV_ERR EV_get_flag_fd (EV_ID *flag_id*, int * *fd*)**Brief**

Obtain fd for polling flag event

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID
----	----------------	-----------------

EV_ID type parameter
1 typedef UINT32 EV_ID

Parameters:

out	<i>fd</i>	int * - event flag queue fd for Polling
-----	-----------	---

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid flag_id

Prerequisite

Execute [EV_create_flag\(\)](#), [EV_create_flag_auto_id\(\)](#) and flag has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

nothing

EV_ERR EV_get_message (EV_ID *queue_id*, EV_Message * *message*)

Brief

Get message event(non-block)

Parameters:

in	<i>queue_id</i>	EV_ID - Message queue ID
----	-----------------	--------------------------

EV_ID type parameter

```
1 typedef UINT32 EV_ID
```

Parameters:

out	<i>message</i>	EV_Message * - Message event structure
-----	----------------	--

EV_Message structure

```
1 typedef unsigned int UINT32;
2 typedef UINT32 EV_ID
3
4 #define EV_MAX_MESSAGE_LENGTH 2048
5 typedef struct {
6   EV_ID queueID; // ID of message queue
7   UINT32 senderInfo; // Message Source
8   UINT32 length; // length of message
9   UINT32 dummy; // dummy for padding
10  UINT8  message[EV_MAX_MESSAGE_LENGTH]; // message
11 } EV_Message;
```

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>queue_id</i>
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_queue\(\)](#), [EV_create_queue_auto_id\(\)](#) and message queue has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

[evk_ioctl](#), [evk_get_event](#)

EV_ERR EV_get_next_event (EV_Event * ev)

Brief get the first reached event(non-block)

Parameters:

out	ev	event Structure
-----	----	-----------------

EV_Event struct

```
1 typedef unsigned int UINT32;
2 typedef UINT32 EV_Type;
3 typedef UINT32 EV_ID;
4 #define EV_MAX_MESSAGE_LENGTH 2048
5 enum ev_message_queue_type {
6  EV_MESSAGE_QUEUE_TYPE_BUSY,
7  EV_MESSAGE_QUEUE_TYPE_FIFO,
8  EV_MESSAGE_QUEUE_TYPE_REPLACE,
9 };
10
11 typedef struct {
12  EV_ID flagID;
13  UINT32 bits;
14 } EV_Flag;
15
16 typedef struct {
17  EV_ID flagID;
18  UINT64 bits;
19 } EV_Flag64;
20
21 typedef struct {
22  EV_ID queueID;
23  UINT32 senderInfo;
24  UINT32 length;
25  UINT32 dummy;
26  UINT8 message[EV_MAX_MESSAGE_LENGTH];
27 } EV_Message;
28
29 typedef struct {
30  EV_Type type; // please reference ev_message_queue_type.
31  union {
32   EV_Flag flag;
33   EV_Flag64 flag64;
34   EV_Message message;
35  } u;
36 } EV_Event;
```

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_Thread</i>	unregistered thread
<i>EV_ERR_Fatal</i>	Fatal error

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ERR EV_get_queue_fd (EV_ID *queue_id*, int * *fd*)

Brief

Obtain fd for polling message event

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID
----	----------------	-----------------

EV_ID type parameter
1 typedef UINT32 EV_ID

Parameters:

out	<i>fd</i>	int * - Message queue fd for Polling
-----	-----------	--------------------------------------

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid queue_id

Prerequisite

Execute [EV_create_queue\(\)](#), [EV_create_queue_auto_id\(\)](#) and message queue has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

nothing

EV_ID EV_moduleID_to_flag64ID (UINT16 *m_id*)

Brief change from moduleID to 64bit flag_id

Parameters:

in	<i>m_id</i>	moduleID
----	-------------	----------

EV_ID type parameter
1 typedef UINT32 EV_ID

Return values:

64bit	flag_id
-------	---------

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ID EV_moduleID_to_flagID (UINT16 *m_id*)

Brief change from moduleID to flag_id

Parameters:

in	<i>m_id</i>	moduleID
----	-------------	----------

Return values:

<i>flag_id</i>	1 typedef UINT32 EV_ID
----------------	------------------------

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ID EV_moduleID_to_queueID (UINT16 *m_id*)**Brief change from moduleID to queue_id****Parameters:**

in	<i>m_id</i>	moduleID
----	-------------	----------

EV_ID type parameter
 1 typedef UINT32 EV_ID

Return values:

<i>queue_id</i>	
-----------------	--

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ERR EV_peek_flag (EV_ID *flag_id*, EV_Flag * *flag*)**Brief get flag event(non-destructive)****Parameters:**

in	<i>flag_id</i>	ID of message event queue
out	<i>flag</i>	flag struct

EV_ID type parameter
 1 typedef UINT32 EV_ID

EV_Flag Structure

```

1 typedef struct {
2   EV_ID flagID;
3   UINT32 bits;
4 } EV_Flag;

```

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_ID</i>	The specified flag ID is not existing, or it is not created in current process.
<i>EV_ERR_Fatal</i>	Fatal error

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ERR EV_peek_flag64 (EV_ID *flag_id*, EV_Flag64 * *flag*)

Brief get flag event(non-destructive)

Parameters:

in	<i>flag_id</i>	ID of message event queue
out	<i>flag</i>	flag struct

EV_ID type parameter
1 typedef UINT32 EV_ID

EV_Flag Structure
1 typedef struct {
2 EV_ID flagID;
3 UINT32 bits;
4 } EV_Flag;

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_ID</i>	The specified flag ID is not existing, or it is not created in current process.

<i>EV_ERR_Fatal</i>	Fatal error
---------------------	-------------

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ERR EV_peek_message (EV_ID *queue_id*, EV_Message * *message*)

Brief get message event(non-destructive)

Parameters:

in	<i>queue_id</i>	ID of message event queue
out	<i>message</i>	message event queue struct

EV_ID define
 1 typedef UINT32 EV_ID

EV_Message struct
 1 #define EV_MAX_MESSAGE_LENGTH 2048
 2 typedef struct {
 3 EV_ID queueID;
 4 UINT32 senderInfo;
 5 UINT32 length;
 6 UINT32 dummy;
 7 UINT8 message[EV_MAX_MESSAGE_LENGTH];
 8 } EV_Message;

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_ID</i>	The specified queue ID is not existing, or it is not created in current process.
<i>EV_ERR_Fatal</i>	Fatal error

Prerequisite

none

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

EV_ERR EV_peek_next_event (EV_Event * ev)

Brief get the first reached event(non-destructive)

Parameters:

out	ev	<p>event Structure</p> <pre>1 typedef unsigned int UINT32; 2 typedef UINT32 EV_Type; 3 typedef UINT32 EV_ID; 4 #define EV_MAX_MESSAGE_LENGTH 2048 5 enum ev_message_queue_type { 6 EV_MESSAGE_QUEUE_TYPE_BUSY, 7 EV_MESSAGE_QUEUE_TYPE_FIFO, 8 EV_MESSAGE_QUEUE_TYPE_REPLACE, 9 }; 10 11 typedef struct { 12 EV_ID flagID; 13 UINT32 bits; 14 } EV_Flag; 15 16 typedef struct { 17 EV_ID flagID; 18 UINT64 bits; 19 } EV_Flag64; 20 21 typedef struct { 22 EV_ID queueID; 23 UINT32 senderInfo; 24 UINT32 length; 25 UINT32 dummy; 26 UINT8 message[EV_MAX_MESSAGE_LENGTH]; 27 } EV_Message; 28 29 typedef struct { 30 EV_Type type; // please reference ev_message_queue_type. 31 union { 32 EV_Flag flag; 33 EV_Flag64 flag64; 34 EV_Message message; 35 } u; 36 } EV_Event;</pre>
-----	----	---

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_Thread</i>	unregistered thread
<i>EV_ERR_Fatal</i>	Fatal error

Prerequisite

none

Change of internal state

none

Classification

Public

Type

Sync Only

EV_ERR_EV_send_message (EV_ID *queue_id*, UINT16 *bytes*, const void * *message*, UINT32 *senderInfo*)

Brief

Send Message event

Parameters:

in	<i>queue_id</i>	EV_ID - Queue ID of the message destination
----	-----------------	---

EV_ID type parameter

1 typedef UINT32 EV_ID

Parameters:

in	<i>bytes</i>	UINT16 - The bytes of the send message
in	<i>message</i>	const void * - Pointer to a transmitting byte line
in	<i>senderInfo</i>	UINT32 - Message Source(It is specified when it is used for application identification. When do not use it, appoint 0.)

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>queue_id</i>
<i>EV_ERR_Busy</i>	Queue overflow
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_queue\(\)](#), [EV_create_queue_auto_id\(\)](#) and message queue has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

evk_open, evk_close, evk_ioctl, evk_store_message

**EV_ERR EV_send_message_by_mID (UINT16 *m_id*, UINT16 *bytes*, const void * *message*,
UINT32 *senderInfo*)**

Brief send message(moduleID specified)**Parameters:**

in	<i>m_id</i>	moduleID
in	<i>bytes</i>	number of send bytes
in	<i>message</i>	pointer to sender contents
in	<i>senderInfo</i>	sender information

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid queue_id
<i>EV_ERR_Busy</i>	Queue overflow
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV create queue\(\)](#),[EV create queue auto id\(\)](#) and message queue has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

evk_open, evk_close, evk_ioctl, evk_store_message

EV_ERR EV_set_flag (EV_ID *flag_id*, UINT32 *bits*)

Brief

Set Event flag

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID of the event flag
----	----------------	-----------------------------------

EV_ID type parameter
 1 typedef UINT32 EV_ID

Parameters:

in	<i>bits</i>	UINT32 - event flag bit pattern
----	-------------	---------------------------------

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid flag_id
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_flag\(\)](#), [EV_create_flag_auto_id\(\)](#) and flag has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

[evk_open](#), [evk_close](#), [evk_ioctl](#), [evk_store_flag](#)

EV_ERR EV_set_flag64 (EV_ID *flag_id*, UINT64 *bits*)**Brief**

Set Event flag

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID of the event flag
----	----------------	-----------------------------------

EV_ID type parameter
 1 typedef UINT32 EV_ID

Parameters:

in	<i>bits</i>	UINT32 - event flag bit pattern
----	-------------	---------------------------------

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid flag_id
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_flag\(\)](#), [EV_create_flag_auto_id\(\)](#) and flag has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

[evk_open](#), [evk_close](#), [evk_ioctl](#), [evk_store_flag](#)

EV_ERR EV_set_flag64_by_mID (UINT16 *m_id*, UINT64 *bits*)**Brief send 64bit flag(moduleID specified)****Parameters:**

in	<i>m_id</i>	moduleID
in	<i>bits</i>	value of flag

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid flag_id
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

none

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

EV_ERR EV_set_flag_by_mID (UINT16 *m_id*, UINT32 *bits*)**Brief send flag(moduleID specified)****Parameters:**

in	<i>m_id</i>	moduleID
----	-------------	----------

in	<i>bits</i>	value of flag
----	-------------	---------------

Return values:

<i>EV_OK</i>	normal termination
<i>EV_ERR_Invalid_ID</i>	set en invalid flag ID
<i>EV_ERR_Fatal</i>	Fatal error

Prerequisite

none

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

EV_ERR EV_wait_flag (EV_ID *flag_id*, EV_Flag * *flag*)

Brief

Get event flag(block)

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID
----	----------------	-----------------

EV_ID type parameter
1 typedef UINT32 EV_ID

Parameters:

out	<i>flag</i>	EV_Flag * - event flag structure
-----	-------------	----------------------------------

EV_Flag structure
1 typedef struct {
2 EV_ID flagID; // Flag ID
3 UINT32 bits; // event flag bit pattern
4 } EV_Flag;

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid flag_id
<i>EV_ERR_Interrupted</i>	signal interrupt
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_flag\(\)](#), [EV_create_flag_auto_id\(\)](#) and flag has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

evk_ioctl, evk_get_event

EV_ERR EV_wait_flag64 (EV_ID *flag_id*, EV_Flag64 * *flag*)

Brief

Get event flag(block)

Parameters:

in	<i>flag_id</i>	EV_ID - Flag ID
----	----------------	-----------------

EV_ID type parameter
1 typedef UINT32 EV_ID

Parameters:

out	<i>flag</i>	EV_Flag * - event flag structure
-----	-------------	----------------------------------

EV_Flag structure
1 typedef unsigned int UINT32;
2 typedef UINT32 EV_ID
3
4 typedef struct {
5 EV_ID flagID; // Flag ID
6 UINT32 bits; // event flag bit pattern
7 } EV_Flag;

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>flag_id</i>
<i>EV_ERR_Interrupted</i>	signal interrupt
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV create flag\(\)](#), [EV create flag_auto id\(\)](#) and *flag* has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

evk_ioctl, evk_get_event

EV_ERR EV_wait_message (EV_ID *queue_id*, EV_Message * *message*)**Brief**

Get message event(block)

Parameters:

in	<i>queue_id</i>	EV_ID - Message queue ID
----	-----------------	--------------------------

EV_ID type parameter
 1 typedef UINT32 EV_ID

Parameters:

out	<i>message</i>	EV_Message * - Message event structure
-----	----------------	--

EV_Message structure

```

1 typedef unsigned int UINT32;
2 typedef UINT32 EV_ID
3
4 #define EV_MAX_MESSAGE_LENGTH 2048
5 typedef struct {
6   EV_ID queueID; // ID of message queue
7   UINT32 senderInfo; // Message Source
8   UINT32 length; // length of message
9   UINT32 dummy; // dummy for padding
10  UINT8 message[EV_MAX_MESSAGE_LENGTH]; // message
11 } EV_Message;
```

Return values:

<i>EV_OK</i>	Successful completion
<i>EV_ERR_Invalid_ID</i>	Invalid <i>queue_id</i>
<i>EV_ERR_Interrupted</i>	signal interrupt
<i>EV_ERR_Fatal</i>	Abnormal error

Prerequisite

Execute [EV_create_queue\(\)](#), [EV_create_queue_auto_id\(\)](#) and message queue has already been created.

Change of internal state

There is no change of the internal Status

Classification

Public

Type

Sync Only

See also:

evk_ioctl, evk_get_event

Posix_based_os001_legacy_library

char * [itoa](#) (int value, char buff[], int radix)uint64_t [ClockCycle](#) (void)unsigned int [delay](#) (unsigned int duration)size_t [strlcat](#) (char *dst, const char *src, size_t siz)size_t [strlcpy](#) (char *dst, const char *src, size_t siz)char * [ultoa](#) (unsigned long value, char *buf, int radix)

Detailed Description

Function Documentation

uint64_t ClockCycle (void)

Brief

Get the number of clock cycles.

Return values:

<i>Microsecond</i>	
--------------------	--

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync

See also:

None

unsigned int delay (unsigned int *duration*)

Brief

Delay for the specified time.

Parameters:

in		
----	--	--

char* itoa (int *value*, char *buff*[], int *radix*)

Brief

Integer value to null terminated string.

Parameters:

in	<i>value</i>	int - value Number to convert
out	<i>buff</i>	char - buff Result string
in	<i>radix</i>	int - radix Radix of value

Return values:

<i>String</i>	pointer
---------------	---------

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync

See also:

None

size_t strcat (char * *dst*, const char * *src*, size_t *siz*)

Brief

Concatenate strings

Parameters:

	<i>[in/out]</i>	dst char * - Destination string
in	<i>src</i>	const char * - Concatenated string
in	<i>siz</i>	size_t - Size of dst

Return values:

<i>String</i>	length after concatenation
---------------	----------------------------

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync

See also:

None

size_t strcpy (char * *dst*, const char * *src*, size_t *siz*)**Brief**

Copy string

Parameters:

	<i>[in/out]</i>	dst char * - Destination string
in	<i>src</i>	const char * - Copy string
in	<i>siz</i>	size_t - Size of dst

Return values:

<i>Length</i>	of copied character string
---------------	----------------------------

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync

See also:

None

char* ultoa (unsigned long *value*, char * *buf*, int *radix*)**Brief**

Convert unsigned long type to NULL terminated character string.

Parameters:

in	<i>value</i>	int - value Number to convert
out	<i>buff</i>	char - buff Result string
in	<i>radix</i>	int - radix Radix of value

Return values:

<i>String</i>	pointer
---------------	---------

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync

See also:

None

Rpc_library

union [RPC_marshall_flag](#)*API call property stored struct. [More...](#)*struct [RPC_marshall_flag.bits](#)**Macros**#define [OTHERSERVICE_RPC_H_](#)#define [RPC_MAX_THREADS_IN_PROCESS](#) 64*RPC_ID max number.*#define [RPC_MAX_FD_IN_PROCESS](#) 130*fd max number in one process. 0 = main/sub thread communication pipe fd. 1 = inotify() process listen fd. 2~129 = even num : API request data socket. odd num : secure stream socket.*#define [RPC_MAX_API_ARG_NUM](#) 8*API call parameter max num.*#define [RPC_MAX_API_ARG_SIZE](#) 1024*API call one parameter max byte.*#define [RPC_MAX_API_ARG_TOTAL_SIZE](#) 1500*API call all parameter max byte.*#define [RPC_MAX_APICALL_QUEUE](#) 16*API call max queue num.*#define [RPC_NO_PORT](#) 0*invalid port num*#define [RPC_NO_ID](#) [RPC_NO_PORT](#)*The ID when destination is invalid.*#define [RPC_SELF_ID](#) [RPC_NO_PORT](#)

```

    The ID when destination is self.
#define RPC\_ANY\_ID RPC\_NO\_PORT
    The ID when destination is any.
#define RPC_DISPATCH_FUNC NULL
#define RPC\_START\_SERVER(ID) RPC_start((ID), RPC_DISPATCH_FUNC, NULL,
NO\_SECURE\_CHECK)
#define RPC\_START\_SECURE\_SERVER(ID) RPC_start((ID), RPC_DISPATCH_FUNC, NULL,
NEED\_SECURE\_CHECK)
#define RPC\_START\_CLIENT(pID) RPC_start((UINT16)RPC\_NO\_ID, NULL, (pID),
NO\_SECURE\_CHECK)
    client start
#define NEED\_SECURE\_CHECK 1
    has secure check
#define NO\_SECURE\_CHECK 0
    has no secure check
#define REGISTERED 1
    registered
#define NO\_REGISTERED 0
    not registered
#define RPC_RETCODE_LEN 9
#define RPC_MARSHALL_FLAG_BITS_CODE 12
#define RPC_MARSHALL_FLAG_BITS_IS_VARARRAY 1
#define RPC_MARSHALL_FLAG_BITS_IS_POINTER 1
#define RPC_MARSHALL_FLAG_BITS_IN_OUT 2
#define rpc_malloc malloc
#define rpc_free free
#define OTHERSERVICE RPCID\_H
    define the RPC_ID used by program like #define XXX_RPC_ID ID but
    RPC\_START\_CLIENT\(\) is not necessary
#define test\_RPC\_ID 49999
    sample(ID is used by RPClibrary test program)
#define MODEMANAGER_RPC_ID 51100 /* ModeManager */
#define ACTIVITYMANAGER_RPC_ID 51101 /* ActivityManager */
#define TIMERENTRYDRV_RPC_ID 53000 /* TimerEntryDrv */
#define TSKM_RPC_ID 53001 /* TaskManager */
#define MSGBRK_RPC_ID 53002 /* MessageBroker */
#define IPMANAGER_RPC_ID 53003 /* IPManager */
#define IPMANAGER_RPC_ID 53003 /* IPManager */
#define DEVICEMANAGER_RPC_ID 53004 /* DeviceManager */
#define _CWORD87_LAN_RPC_ID 53005 /* Communication(_CWORD87_LAN) */
#define CAN_RPC_ID 53006 /* Communication(CAN) */
#define _CWORD83_RPC_ID 53007 /* Communication(_CWORD83_) */
#define SENSOR_RPC_ID 53008 /* Vehicle(Sensor) */
#define GPS_RPC_ID 53009 /* Vehicle(GPS) */

```

```
#define RESMGR_RPC_ID 53010 /* ResourceManager */
#define GRAPHICS_RPC_ID 53011 /* Graphic */
#define OTHERSERVICE_RPCLIBRARY_H_
```

Typedefs

```
typedef INT32 RPC Result
```

RPC library function return type.

```
typedef UINT32 RPC ID
```

RPC ID.

```
typedef RPC Result(* RPC_dispatch_func_t) (UINT16 api_num, const char *args_string, unsigned  
int args_size, char **ret_string, unsigned int *ret_bytes)
```

Enumerations

```
enum rpc\_result { RPC\_OK = ***, RPC\_ERR No Response = ***, RPC\_ERR Timeout,  
RPC\_ERR Busy, RPC\_ERR API Error, RPC\_ERR API Fatal, RPC\_ERR Fatal,  
RPC\_ERR Configuration, RPC\_ERR Server DeadLock, RPC\_ERR Server Finish,  
RPC\_ERR Reject connect }RPC library function return value.
```

Functions

```
RPC Result RPC_start (RPC ID id, RPC\_dispatch\_func\_t func, RPC ID *pID, INT32 secure_check)
```

```
RPC Result RPC\_get\_fd (RPC ID id, int *fd)
```

```
RPC Result RPC\_process API request (RPC ID id)
```

```
void RPC\_end (RPC ID id)
```

```
void RPC\_end\_all (void)
```

```
RPC Result RPC\_set API timeout (INT32 sec)
```

```
RPC Result RPC\_is\_server\_ready (RPC ID id)
```

```
RPC Result RPC\_regist\_credential (int uid_num, uid_t *uid_list, int gid_num, gid_t *gid_list)
```

```
RPC Result RPC\_get\_client\_credential (uid_t *client_uid, gid_t *client_gid)
```

```
RPC Result RPC\_API\_call (RPC ID id, UINT16 api_num, const char *args_string, unsigned int  
args_size, char **ret_string, unsigned int *ret_size)
```

```
int RPC\_demarshal\_arguments (const char *from, unsigned int size, int need_alloc, int  
num_args,...)
```

```
char * RPC\_marshall\_arguments (unsigned int *size, int dont_marshall_out_args, int num_args,...)
```

```
void RPC\_marshall\_free (int num,...)
```

```
void RPC\_free\_return\_string (void *ptr)
```

Variables

```
char * RPC_DEBUG
```

Detailed Description

Class Documentation

union RPC_marshall_flag

APIcall property stored struct.

Class Members:

UINT32	all	all property
struct RPC_marshall_flag	bits	variable properties

struct RPC_marshall_flag.bits

Class Members:

UINT16	bytes	string or user define type byte num.other is 0
unsigned int	code: RPC_MARSHALL_FLAG_BITS_CODE	value type
unsigned int	in_out: RPC_MARSHALL_FLAG_BITS_IN_OUT	in/out parameter property
unsigned int	is_pointer: RPC_MARSHALL_FLAG_BITS_IS_POINTER	pointer or not
unsigned int	is_vararray: RPC_MARSHALL_FLAG_BITS_IS_VARARRAY	vararray or not

Macro Definition Documentation

#define OTHERSERVICE_RPCID_H_

define the RPC_ID used by program like #define XXX_RPC_ID ID but [RPC_START_CLIENT\(\)](#) is not necessary

RPC_ID as the following

- 1 UI Application : 50xxx
- 2 Application Service : 51xxx
 - HMI : 511xx
 - AV : 512xx
 - Radio : 513xx
 - Connectivity : 514xx
 - Navigation/Location: 515xx

- Vehicle : 516xx
- Communication : 517xx
- BasePF : 518xx
- System : 519xx
- : 510xx
- 3 Common Service : 53xxx
- 4 Extension Service : 55xxx
- 5 System/Device Driver : 56xxx
- 6 4915259999 are Reserved

7 **Note:**

RPC_ID==port num , when not call from RPClibrary, and not well-known(~1023)/registered(102449151) use the fixed port num to IP communicate, use 6xxx port number.
 when RPC_START_CLIENT,and distribute automatically. use the linux kernel port distribute function 10244999 as the port num unused.

8

#define RPC_START_CLIENT(pID) RPC_start((UINT16)[RPC_NO_ID](#), NULL, (pID), [NO_SECURE_CHECK](#))

client start

Brief

client start

Parameters:

out	<i>pID</i>	RPC_ID * -auto allocated program's RPC_ID
-----	------------	---

Return values:

<i>RPC_OK</i>	Normal End
<i>RPC_ERR_Fatal</i>	Fatal Error

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

RPC_start, [RPC_end](#)

#define RPC_START_SECURE_SERVER(ID) RPC_start((ID), RPC_DISPATCH_FUNC, NULL, [NEED SECURE CHECK](#))

Brief

server start(certification of the client by UID)

Parameters:

in	<i>ID</i>	RPC_ID - program's RPC_ID
----	-----------	---------------------------

Return values:

<i>RPC_OK</i>	Normal End
<i>RPC_ERR_Fatal</i>	Fatal Error

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

RPC_start, [RPC_end](#)

#define RPC_START_SERVER(ID) RPC_start((ID), RPC_DISPATCH_FUNC, NULL, [NO SECURE CHECK](#))

Brief

server start(certification of the client by UID)

Parameters:

in	<i>ID</i>	RPC_ID - program's RPC_ID
----	-----------	---------------------------

Return values:

<i>RPC_OK</i>	Normal End
<i>RPC_ERR_Fatal</i>	Fatal Error

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

RPC_start, [RPC_end](#)

Typedef Documentation**typedef UINT32 [RPC ID](#)**

RPC ID.

typedef INT32 [RPC Result](#)

RPC library function return type.

See also:

[rpc_result](#)

Enumeration Type Documentation**enum [rpc_result](#)**

RPC library function return value.

Enumerator

RPC_OK ok

RPC_ERR_No_Response has no response

RPC_ERR_Timeout timeout

RPC_ERR_Busy busy

RPC_ERR_API_Error API error.

RPC_ERR_API_Fatal API fatal.

RPC_ERR_Fatal fatal

RPC_ERR_Configuration configuration

RPC_ERR_Server_DeadLock server deadlock

RPC_ERR_Server_Finish server finish

RPC_ERR_Reject_connect reject connect

Function Documentation

RPC Result `RPC_API_call (RPC_ID id, UINT16 api_num, const char * args_string, unsigned int args_size, char ** ret_string, unsigned int * ret_size)`

brief

API call execution function (client)

Parameters:

in	<i>id</i>	RPC_ID - RPC_ID of server program
in	<i>api_num</i>	UINT16 - Request API number
in	<i>args_string</i>	const char* - Argument string
in	<i>args_size</i>	unsigned int - Argument string length
out	<i>ret_string</i>	char** - Process result string
out	<i>ret_size</i>	unsigned int* - Process result string length

Return values:

<i>RPC_OK</i>	Normal End
<i>RPC_ERR_Fatal</i>	Fatal error
<i>RPC_ERR_No_Response</i>	No response
<i>RPC_ERR_Busy</i>	API call queue overflow
<i>RPC_ERR_Server_DeadLock</i>	RPC server deadlock
<i>RPC_ERR_Reject_connect</i>	The RPC server does not allow communication

Prerequisite

The RPC server is running.

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

int `RPC_demarshal_arguments (const char * from, unsigned int size, int need_alloc, int num_args, ...)`

Brief

Confirm whether the input character string conforms to the prescribed format.

Parameters:

in	<i>from</i>	const char * - Input string
in	<i>size</i>	unsigned int - Byte length of the input string
in	<i>need_alloc</i>	int - Flag whether memory allocation is necessary or not.
in	<i>num_args</i>	int - Number of the variable length arguments
in	...	void * - Variable length arguments

Return values:

0	Normal End
-1	Abnormal End

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:[RPC marshall arguments](#)**void RPC_end (RPC ID *id*)****Brief**

End processing of RPC library.

Parameters:

in	<i>id</i>	RPC_ID - program's RPC_ID
----	-----------	---------------------------

Return values:

<i>None</i>

Prerequisite

RPC_START_SERVER or RPC_START_SECURE_SERVER or RPC_START_CLIENT is already called.

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

void RPC_end_all (void)

Brief

End processing of RPC library(forcing to clean up).

Prerequisite

-None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

void RPC_free_return_string (void * ptr)

Brief

release memory allocated during running API call.

Parameters:

in	<i>ptr</i>	void * - allocated memory
----	------------	---------------------------

Return values:

<i>None</i>

Prerequisite

RPC_API_call is already called.

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

[RPC Result](#) `RPC_get_client_credential (uid_t * client_uid, gid_t * client_gid)`

Brief

Get UID and GID of client that requested to run APIs processing

Parameters:

out	<i>client_uid</i>	uid_t* - UID of client that requested to run APIs processing
out	<i>client_gid</i>	gid_t* - GID of client that requested to run APIs processing

Return values:

<i>RPC_OK</i>	Normal End
<i>RPC_ERR_Fatal</i>	Fatal Error

Prerequisite

`RPC_START_SECURE_SERVER` is already called.

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

[RPC Result](#) `RPC_get_fd (RPC_ID id, int * fd)`

Brief

Get the file descriptor to judge whether there is the request of RPC-API call or not.

Parameters:

in	<i>id</i>	RPC_ID - server program's RPC_ID
out	<i>fd</i>	int * - area for fd

Return values:

<i>RPC_OK</i>	Normal End
<i>RPC_ERR_Fatal</i>	Fatal Error

Prerequisite

`RPC_START_SERVER` or `RPC_START_SECURE_SERVER` is already called.

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

RPC Result `RPC_is_server_ready` (RPC ID *id*)**Brief**

Return RPC server's status

Parameters:

in	<i>id</i>	RPC_ID - server program's RPC_ID
----	-----------	----------------------------------

Return values:

<i>RPC_OK</i>	: Success
<i>RPC_ERR_No_Response</i>	: No response

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

`char* RPC_marshall_arguments` (`unsigned int * size`, `int dont_marshall_out_args`, `int num_args`, ...)**Brief**

Collect variable arguments, convert them to a format that matches the receiver's specification, and stuff them into the message buffer.

Parameters:

in	<i>size</i>	unsigned int * - data size after conversion
in	<i>dont_marshall_out_args</i>	int - input/output kind(when the argument is a pointer)
in	<i>num_args</i>	int - value of variable argument
in	...	void * - variable argument

Return values:

<i>char</i>	* Message buffer after conversion
-------------	-----------------------------------

<i>NULL</i>	Abnormal
-------------	----------

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

RPC_demarshalol_arguments

void RPC_marshall_free (int *num*, ...)

Brief

Variadic function for memory release

Parameters:

in	<i>num</i>	int -number of memory to be released
in	...	void * - allocated memory

Return values:

<i>None</i>

Prerequisite

None

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

[RPC Result](#) RPC_process_API_request ([RPC ID](#) *id*)

Brief

Run the processing of RPC-API call

Parameters:

in	<i>id</i>	RPC_ID - server program's RPC_ID
----	-----------	----------------------------------

Return values:

<i>RPC_OK</i>	Normal End
<i>RPC_ERR_Fatal</i>	Fatal Error

Prerequisite

RPC_START_SERVER or RPC_START_SECURE_SERVER is already called.

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

[RPC Result](#) `RPC_regist_credential (int uid_num, uid_t * uid_list, int gid_num, gid_t * gid_list)`

Brief

Register UID and GID list

Parameters:

in	<i>uid_num</i>	int - UID number to regist(number of elements of UID list)
in	<i>uid_list</i>	uid_t* - head pointer of UID list
in	<i>gid_num</i>	int - GID number to regist(number of elements of GID list)
in	<i>gid_list</i>	gid_t* - head pointer of GID list

Return values:

<i>RPC_OK</i>	Normal End
<i>RPC_ERR_Fatal</i>	Fatal Error
<i>RPC_ERR_Configuration</i>	Argument is out of range

Prerequisite

RPC_START_SECURE_SERVER is already called.

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

None

[RPC Result](#) `RPC_set_API_timeout (INT32 sec)`

Brief

Set timeout time for server side API processing time

Parameters:

in	sec	INT32 - timeout time
----	-----	----------------------

Return values:

<i>RPC_OK</i>	Normal End
<i>RPC_ERR_Fatal</i>	Fatal Error
<i>RPC_ERR_Configuration</i>	Argument is out of range

Prerequisite

`RPC_START_SERVER` or `RPC_START_SECURE_SERVER` is already called.

Change of internal state

None

Classification

Public

Type

Sync Only

See also:

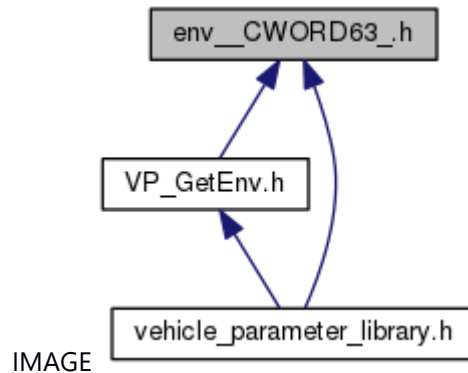
None

File Documentation

env__CWORD63_.h File Reference

Define of *CWORD63* vehicle parameter environment variable

This graph shows which files directly or indirectly include this file:



Macros

```
#define VP_CWORD63_VERSION "VP_CWORD63_version"  
#define VP_CWORD63_CWORD63_ID "VP_CWORD63__CWORD63_ID"  
#define VP_CWORD63_MAKER_ID "VP_CWORD63_maker_ID"  
#define VP_CWORD63_AREA "VP_CWORD63_area"  
#define VP_CWORD63_GRADE "VP_CWORD63_grade"  
#define VP_CWORD63_MODEL_ID "VP_CWORD63_model_ID"  
#define VP_CWORD63_DEST_C_CODE "VP_CWORD63_dest_c_code"  
#define VP_CWORD63_RADIO_TUNER_NUM "VP_CWORD63_radio_tuner_num"  
#define VP_CWORD63_RADIO_ANTENNA_NUM "VP_CWORD63_radio_antenna_num"  
#define VP_CWORD63_DAB_TUNER_NUM "VP_CWORD63_DAB_tuner_num"  
#define VP_CWORD63_DAB_SUB_ANTENNA "VP_CWORD63_DAB_sub_antenna"  
#define VP_CWORD63_CWORD32_DAB "VP_CWORD63__CWORD32_DAB"  
#define VP_CWORD63_CACHERADIO "VP_CWORD63_CacheRadio"  
#define VP_CWORD63_AMP_TYPE "VP_CWORD63_AMP_type"  
#define VP_CWORD63_OP_COOPERATION "VP_CWORD63_OP_cooperation"  
#define VP_CWORD63_DISPLAY "VP_CWORD63_DISPLAY"  
#define VP_CWORD63_TOUCHPANEL "VP_CWORD63_TOUCHPANEL"  
#define VP_CWORD63_DECK "VP_CWORD63_DECK"  
#define VP_CWORD63_CWORD105 "VP_CWORD63__CWORD105_"  
#define VP_CWORD63_DEFAULT_MODE "VP_CWORD63_default_mode"  
#define  
    VP_CWORD63_VOICE_RECOGNITION_TUTORIAL "VP_CWORD63_VOICE_RECOGNITION_T  
    UTORIAL"  
#define VP_CWORD63_SEARCH "VP_CWORD63_SEARCH"  
#define VP_CWORD63_CONTACT "VP_CWORD63_CONTACT"
```

```

#define VP\_CWORD63\_JPTRAFFICINFO\_SHOW "VP_CWORD63_JPTrafficInfo_show"
#define
    VP\_CWORD63\_INFOMATION\_MANAGEMENT "VP_CWORD63_INFOMATION_MANAGEMENT"
#define VP\_CWORD63\_SUBSCREEN\_APPS\_SW "VP_CWORD63_SUBSCREEN_APPS_SW"
#define VP\_CWORD63\_WIFI\_STA\_ENABLE "VP_CWORD63_WIFI_STA_ENABLE"
#define VP\_CWORD63\_WIFI\_AP\_ENABLE "VP_CWORD63_WIFI_AP_ENABLE"
#define VP\_CWORD63\_WIFI\_DIRECT\_ENABLE "VP_CWORD63_WIFI_DIRECT_ENABLE"
#define VP\_CWORD63\_HAS\_GPS "VP_CWORD63_HAS_GPS"
#define
    VP\_CWORD63\_HAS\_CWORD80\_DEFAULT "VP_CWORD63_HAS_CWORD80_DEFAULT"
#define
    VP\_CWORD63\_DISCONNECT\_CWORD80\_TIMEOUT "VP_CWORD63_DISCONNECT_CWORD80_TIMEOUT"
#define VP\_CWORD63\_PHASE "VP_CWORD63_PHASE"
#define VP\_CWORD63\_CERTIFICATION\_SOFT "VP_CWORD63_CERTIFICATION_SOFT"
#define VP\_CWORD63\_CWORD6 "VP_CWORD63_CWORD6_"
#define VP\_CWORD63\_VOICE\_TAGS "VP_CWORD63_VOICE_TAGS"
#define VP\_CWORD63\_BT\_2HFP "VP_CWORD63_BT_2HFP"
#define VP\_CWORD63\_CONNECTION\_SERVER "VP_CWORD63_CONNECTION_SERVER"
#define
    VP\_CWORD63\_CWORD33\_LOG\_ENC\_KEYTYPE "VP_CWORD63_CWORD33_LOG_ENC_KEYTYPE"
#define VP\_CWORD63\_DES\_VOL\_1 "VP_CWORD63_DES_VOL_1"
#define VP\_CWORD63\_DES\_VOL\_2 "VP_CWORD63_DES_VOL_2"
#define VP\_CWORD63\_DES\_VOL\_3 "VP_CWORD63_DES_VOL_3"
#define VP\_CWORD63\_DES\_VOL\_4 "VP_CWORD63_DES_VOL_4"
#define VP\_CWORD63\_DES\_VOL\_5 "VP_CWORD63_DES_VOL_5"
#define VP\_CWORD63\_DES\_VOL\_6 "VP_CWORD63_DES_VOL_6"
#define VP\_CWORD63\_DES\_VOL\_7 "VP_CWORD63_DES_VOL_7"
#define VP\_CWORD63\_DES\_VOL\_8 "VP_CWORD63_DES_VOL_8"
#define VP\_CWORD63\_STEERING\_TYPE "VP_CWORD63_STEERING_TYPE"
#define
    VP\_CWORD63\_HANDWRITEMODE\_MOVERANGE "VP_CWORD63_Handwritemode_moverange"
#define VP\_CWORD63\_TOUCHPAD\_HOME\_SW "VP_CWORD63_TOUCHPAD_HOME_SW"
#define VP\_CWORD63\_CWORD17 "VP_CWORD63_CWORD17_"
#define VP\_CWORD63\_BT\_ADD\_DEVICE\_SW "VP_CWORD63_BT_ADD_DEVICE_SW"
#define
    VP\_CWORD55\_CWORD66\_SVR\_PRODUCT\_DOWNLOAD\_URL "VP_CWORD55_CWORD66_SVR_PRODUCT_DOWNLOAD_URL"
#define
    VP\_CWORD55\_CWORD66\_SVR\_TRY\_DOWNLOAD\_URL "VP_CWORD55_CWORD66_SVR_TRY_DOWNLOAD_URL"
#define
    VP\_CWORD55\_CWORD66\_SVR\_TEST\_DOWNLOAD\_URL "VP_CWORD55_CWORD66_SVR_TEST_DOWNLOAD_URL"

```

```

#define
  VP_CWORD55_CWORD66_SVR_PRODUCT_UPLOAD_URL "VP_CWORD55_CWORD66_S
  VR_PRODUCT_UPLOAD_URL"
#define
  VP_CWORD55_CWORD66_SVR_TRY_UPLOAD_URL "VP_CWORD55_CWORD66_SVR_TR
  Y_UPLOAD_URL"
#define
  VP_CWORD55_CWORD66_SVR_TEST_UPLOAD_URL "VP_CWORD55_CWORD66_SVR_T
  EST_UPLOAD_URL"
#define
  VP_CWORD55_CWORD66_SVR_PRODUCT_BROWSER_URL "VP_CWORD55_CWORD66_
  _SVR_PRODUCT_BROWSER_URL"
#define
  VP_CWORD55_CWORD66_SVR_TRY_BROWSER_URL "VP_CWORD55_CWORD66_SVR_
  TRY_BROWSER_URL"
#define
  VP_CWORD55_CWORD66_SVR_TEST_BROWSER_URL "VP_CWORD55_CWORD66_SVR_
  TEST_BROWSER_URL"
#define
  VP_CWORD55_CWORD66_SVR_PRODUCT_DOWNLOAD_HOST "VP_CWORD55_CWOR
  D66_SVR_PRODUCT_DOWNLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_TRY_DOWNLOAD_HOST "VP_CWORD55_CWORD66_S
  VR_TRY_DOWNLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_TEST_DOWNLOAD_HOST "VP_CWORD55_CWORD66_
  SVR_TEST_DOWNLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_PRODUCT_UPLOAD_HOST "VP_CWORD55_CWORD66_
  _SVR_PRODUCT_UPLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_TRY_UPLOAD_HOST "VP_CWORD55_CWORD66_SVR_
  TRY_UPLOAD_HOST"
#define
  VP_CWORD55_CWORD66_SVR_TEST_UPLOAD_HOST "VP_CWORD55_CWORD66_SVR
  _TEST_UPLOAD_HOST"
#define
  VP_CWORD55_REPRO_SVR_PRODUCT_URL "VP_CWORD55_REPRO_SVR_PRODUCT_URL"
#define VP_CWORD55_REPRO_SVR_TRY_URL "VP_CWORD55_REPRO_SVR_TRY_URL"
#define VP_CWORD55_REPRO_SVR_TEST_URL "VP_CWORD55_REPRO_SVR_TEST_URL"
#define VP_CWORD63_CWORD70_EU "VP_CWORD63_CWORD70_EU"
#define VP_CWORD63_CWORD40 "VP_CWORD63_CWORD40_"
#define VP_CWORD63_DCM_EC "VP_CWORD63_DCM_EC"
#define
  VP_CWORD63_EXTERNALDISPLAY_INTERACTION "VP_CWORD63_ExternalDisplay_Inte
  raction"
#define VP_CWORD63_DES_OUTPUT_FL_SP "VP_CWORD63_DES_OUTPUT_FL_SP"
#define VP_CWORD63_DES_OUTPUT_FR_SP "VP_CWORD63_DES_OUTPUT_FR_SP"

```

```

#define VP_CWORD63_DES_OUTPUT_RL_SP "VP_CWORD63_DES_OUTPUT_RL_SP"
#define VP_CWORD63_DES_OUTPUT_RR_SP "VP_CWORD63_DES_OUTPUT_RR_SP"
#define VP_CWORD63_DES_OUTPUT_FC_SP "VP_CWORD63_DES_OUTPUT_FC_SP"
#define VP_CWORD63_DES_OUTPUT_ML_SP "VP_CWORD63_DES_OUTPUT_ML_SP"
#define VP_CWORD63_DES_OUTPUT_MR_SP "VP_CWORD63_DES_OUTPUT_MR_SP"
#define VP_CWORD63_DES_OUTPUT_NAVI_SP "VP_CWORD63_DES_OUTPUT_NAVI_SP"
#define VP_CWORD63_DES_OUTPUT_HRDRIVER "VP_CWORD63_DES_OUTPUT_HRDriver"
#define
    VP_CWORD63_DES_OUTPUT_HRPASSENGER "VP_CWORD63_DES_OUTPUT_HRPasseng
    er"
#define VP_CWORD63_AUTOMATED_DRIVING "VP_CWORD63_AUTOMATED_DRIVING"
#define VP_CWORD63_OPN_MOV "VP_CWORD63_OPN_MOV"
#define VP_CWORD63_CWORD84_SDSLOT "VP_CWORD63_CWORD84_SDslot"
#define VP_CWORD63_BTANT_TYPE "VP_CWORD63_BTANT_TYPE"
#define VP_CWORD63_GVIF_FROM_CWORD80_ "VP_CWORD63_GVIF_FROM_CWORD80_"
#define VP_CWORD63_MIC_MFR "VP_CWORD63_MIC_MFR"

```

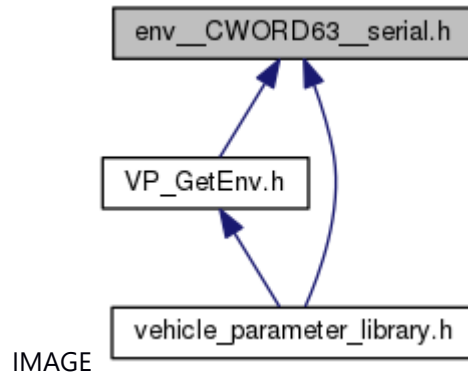
Detailed Description

Define of CWORD63 vehicle parameter environment variable

env__CWORD63__serial.h File Reference

Define of *CWORD63* (Serial) vehicle parameter environment variable

This graph shows which files directly or indirectly include this file:



Macros

```
#define VP\_CWORD63\_SER\_SERIAL "VP_CWORD63_SER_serial"  
#define VP\_CWORD63\_SER\_CWORD49\_ID "VP_CWORD63_SER_CWORD49_ID"  
#define VP\_CWORD63\_SER\_HELPNET\_ID "VP_CWORD63_SER_HELPNET_ID"  
#define VP\_CWORD63\_SER\_MAP\_UPDATE\_ID "VP_CWORD63_SER_map_update_ID"  
#define VP\_CWORD63\_SER\_SHIP\_DATE "VP_CWORD63_SER_ship_date"  
#define VP\_CWORD63\_SER\_FACTORY\_ID "VP_CWORD63_SER_FACTORY_ID"
```

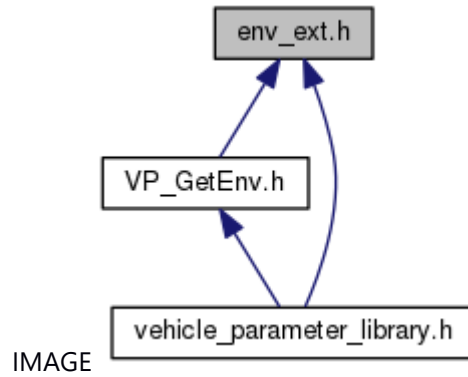
Detailed Description

Define of *CWORD63* (Serial) vehicle parameter environment variable

env_ext.h File Reference

Define of supplier vehicle parameter environment variable.

This graph shows which files directly or indirectly include this file:



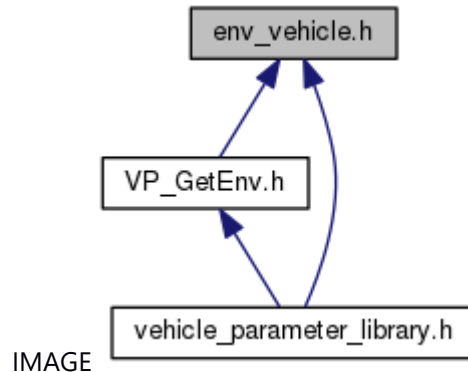
Detailed Description

Define of supplier vehicle parameter environment variable.

env_vehicle.h File Reference

Define of *CWORD31* vehicle parameter environment variable

This graph shows which files directly or indirectly include this file:



Macros

```
#define VP_CWORD31_VERSION "VP_CWORD31_version"  
#define VP_CWORD31_CLASS_TYPE_OF_LAN "VP_CWORD31_class_type_of_LAN"  
#define VP_CWORD31_REVERSE "VP_CWORD31_reverse"  
#define VP_CWORD31_PKB "VP_CWORD31_PKB"  
#define VP_CWORD31_ADIM_CWORD62 "VP_CWORD31_ADIM_CWORD62_  
#define VP_CWORD31_TELEMATICS_FUNCTION "VP_CWORD31_telematics_function"  
#define VP_CWORD31_OPENING_PICTURE "VP_CWORD31_opening_picture"  
#define VP_CWORD31_THEME_COLOR "VP_CWORD31_theme_color"  
#define VP_CWORD31_AC_ONSCREEN "VP_CWORD31_AC_onscreen"  
#define VP_CWORD31_AC_SCREEN_OPERATION "VP_CWORD31_AC_screen_operation"  
#define VP_CWORD31_AC_SCREEN_HOME "VP_CWORD31_AC_screen_home"  
#define VP_CWORD31_AC_SWITCH_FRONT_TEMP "VP_CWORD31_AC_switch_front_temp"  
#define VP_CWORD31_AC_TEMP_DISP_OUTSIDE "VP_CWORD31_AC_temp_disp_outside"  
#define VP_CWORD31_AC_TEMP_DISP_SETTING "VP_CWORD31_AC_temp_disp_setting"  
#define VP_CWORD31_AC_SWITCH_CONCIERGE "VP_CWORD31_AC_switch_concierge"  
#define VP_CWORD31_AC_SWITCH_S_FLOW "VP_CWORD31_AC_switch_S_flow"  
#define VP_CWORD31_CWORD11 "VP_CWORD31__CWORD11_  
#define VP_CWORD31_CWORD11_HOME "VP_CWORD31__CWORD11_home"  
#define  
    VP_CWORD31_CWORD11_DRIVE_WHEEL "VP_CWORD31__CWORD11_drive_wheel"  
#define VP_CWORD31_CWORD11_DRIVE_TYPE "VP_CWORD31__CWORD11_drive_type"  
#define  
    VP_CWORD31_CWORD11_BATTERY_POSITION "VP_CWORD31__CWORD11_battery_po  
    sition"  
#define  
    VP_CWORD31_CWORD11_BATTERY_DIRECTION "VP_CWORD31__CWORD11_battery_d  
    irection"
```

```

#define
    VP_CWORD31_CWORD11_MOTOR_POSITION "VP_CWORD31__CWORD11__motor_posi
tion"
#define VP_CWORD31_CWORD9_TRIP "VP_CWORD31__CWORD9__trip"
#define VP_CWORD31_CWORD9_HOME_TRIP "VP_CWORD31__CWORD9__home_trip"
#define VP_CWORD31_CWORD9_FUEL "VP_CWORD31__CWORD9__fuel"
#define VP_CWORD31_CWORD9_HOME_FUEL "VP_CWORD31__CWORD9__home_fuel"
#define
    VP_CWORD31_FC_ECO_SAFETY_POINT_SERVICE "VP_CWORD31__FC__eco_safety_point_ser
vice"
#define VP_CWORD31_FC_SCALE_GAS "VP_CWORD31__FC__scale_GAS"
#define VP_CWORD31_FC_SCALE_HV "VP_CWORD31__FC__scale_HV"
#define VP_CWORD31_FC_ENERGY_MARK "VP_CWORD31__FC__energy_mark"
#define VP_CWORD31_CWORD9_TYPE "VP_CWORD31__CWORD9__type"
#define VP_CWORD31_FC_MET_CWORD90_04 "VP_CWORD31__FC__MET__CWORD90_04"
#define VP_CWORD31_DISPLAY_SETTING_SCREEN "VP_CWORD31__display_setting_screen"
#define VP_CWORD31_CWORD49_BRAND "VP_CWORD31__CWORD49__brand"
#define VP_CWORD31_VR_HVAC_CONTROL "VP_CWORD31__VR__HVAC_control"
#define VP_CWORD31_CLOCK_TYPE "VP_CWORD31__clock_type"
#define VP_CWORD31_START_UP_BGM "VP_CWORD31__start_up_BGM"
#define VP_CWORD31_START_UP_BGM_TYPE "VP_CWORD31__start_up_BGM_type"
#define VP_CWORD31_DTV_ANTENNA_FL "VP_CWORD31__DTV__antenna_FL"
#define VP_CWORD31_DTV_ANTENNA_FR "VP_CWORD31__DTV__antenna_FR"
#define VP_CWORD31_DTV_ANTENNA_RL "VP_CWORD31__DTV__antenna_RL"
#define VP_CWORD31_DTV_ANTENNA_RR "VP_CWORD31__DTV__antenna_RR"
#define VP_CWORD31_HF_VOICE_QUALITY_TYPE "VP_CWORD31__HF__voice_quality_type"
#define VP_CWORD31_TUNE_KNOB_PUSH "VP_CWORD31__tune_knob_push"
#define VP_CWORD31_SCREEN_SWITCH_SEEK "VP_CWORD31__screen_switch_seek"
#define VP_CWORD31_DAB_ANTENNA "VP_CWORD31__DAB__antenna"
#define VP_CWORD31_HARD_SWITCH_BEEP "VP_CWORD31__hard_switch_beep"
#define VP_CWORD31_SCREEN_SWITCH_BEEP "VP_CWORD31__screen_switch_beep"
#define VP_CWORD31_STEERING_SWITCH_TYPE "VP_CWORD31__steering_switch_type"
#define VP_CWORD31_STEERING_SWITCH_TEL "VP_CWORD31__steering_switch_tel"
#define VP_CWORD31_CLASS_CIRCUIT_MODE "VP_CWORD31__class_circuit_mode"
#define VP_CWORD31_CWORD67_WARNING "VP_CWORD31__CWORD67__warning"
#define VP_CWORD31_CWORD67_ACCELERATOR "VP_CWORD31__CWORD67__accelerator"
#define VP_CWORD31_PANEL_MICROCOMPUTER "VP_CWORD31__panel_microcomputer"
#define
    VP_CWORD31_CLASS_WATCH_AT_BRAKE_HOLD "VP_CWORD31__class_watch_at_brake_ho
ld"
#define VP_CWORD31_GPS_ANTENNA "VP_CWORD31__GPS__antenna"
#define VP_CWORD31_VEHICLE_SIGNAL_ILL "VP_CWORD31__vehicle_signal_ILL"
#define VP_CWORD31_SWITCH_TRIGGER "VP_CWORD31__switch_trigger"
#define VP_CWORD31_SENSITIVITY_LEVEL "VP_CWORD31__sensitivity_level"
#define VP_CWORD31_INVALIDITY_TERM "VP_CWORD31__invalidity_term"
#define VP_CWORD31_THUMBNAIL_TYPE "VP_CWORD31__thumbnail_type"
#define VP_CWORD31_BT_DEVICE_NAME "VP_CWORD31__BT__device_name"
#define VP_CWORD31_ECO_DRIVE_MAX_VOLUME "VP_CWORD31__eco_drive_max_volume"

```



```
#define VP\_CWORD31\_EOM\_FUNCTION "VP_CWORD31_EOM_function"  
#define VP\_CWORD31\_MICROPHONE\_TYPE "VP_CWORD31_Microphone_Type"  
#define VP\_CWORD31\_MICROPHONE\_LOCATION "VP_CWORD31_Microphone_Location"  
#define VP\_CWORD31\_WIDE\_BAND\_FM "VP_CWORD31_Wide_band_FM"  
#define VP_CWORD31_MAP_COLOR "VP_CWORD31_map_color"  
#define VP_CWORD31_CLOUD_ECO "VP_CWORD31_cloud_eco"  
#define VP_CWORD31_EG_TYPE "VP_CWORD31_EG_type"  
#define VP\_CWORD31\_DISPLAY\_TYPE "VP_CWORD31_display_type"  
#define VP\_CWORD31\_CLOCK\_DISPLAY "VP_CWORD31_clock_display"  
#define VP\_CWORD31\_DESTINATION "VP_CWORD31_destination"  
#define VP\_CWORD31\_CLASS\_OPERATION "VP_CWORD31_class_operation"  
#define VP\_CWORD31\_OPENING\_INTERVAL "VP_CWORD31_opening_interval"  
#define VP\_CWORD31\_DES\_FUNCTION "VP_CWORD31_DES_function"  
#define VP\_CWORD31\_BEEP\_SOUND\_KEYS "VP_CWORD31_beep_sound_keys"
```

Detailed Description

Define of *CWORD31* vehicle parameter environment variable

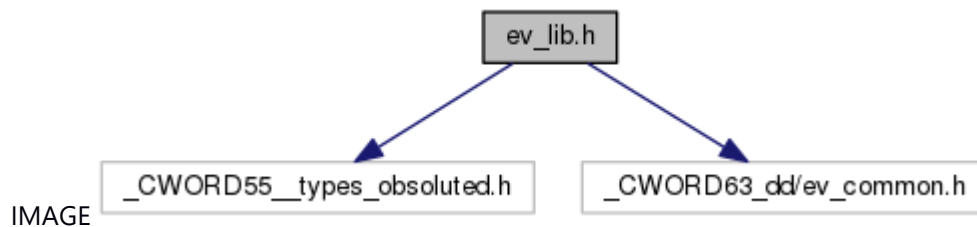
ev_lib.h File Reference

Event library API define head file.

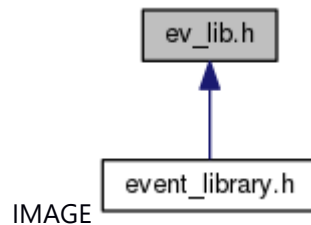
```
#include <_CWORD55__types_obsoluted.h>
```

```
#include <_CWORD63_dd/ev_common.h>
```

Include dependency graph for ev_lib.h:



This graph shows which files directly or indirectly include this file:



Functions

EV_ERR [EV_create_flag](#) (EV_ID flag_id)

EV_ERR [EV_create_flag64](#) (EV_ID flag_id)

EV_ERR [EV_create_queue](#) (EV_ID queue_id, UINT8 length, UINT16 max_bytes, EV_Message_Queue_Type type)

EV_ERR [EV_create_flag_auto_id](#) (EV_ID *flag_id)

EV_ERR [EV_create_flag64_auto_id](#) (EV_ID *flag_id)

EV_ERR [EV_create_queue_auto_id](#) (EV_ID *queue_id, UINT8 length, UINT16 max_bytes, EV_Message_Queue_Type type)

EV_ERR [EV_destroy_flag](#) (EV_ID queue_id)

EV_ERR [EV_destroy_queue](#) (EV_ID queue_id)

EV_ERR [EV_set_flag](#) (EV_ID flag_id, UINT32 bits)

EV_ERR [EV_set_flag64](#) (EV_ID flag_id, UINT64 bits)

EV_ERR [EV_send_message](#) (EV_ID queue_id, UINT16 bytes, const void *message, UINT32 senderInfo)

EV_ERR [EV_get_next_event](#) (EV_Event *ev)

EV_ERR [EV_peek_next_event](#) (EV_Event *ev)

EV_ERR [EV_get_flag](#) (EV_ID flag_id, EV_Flag *flag)

EV_ERR [EV_wait_flag](#) (EV_ID flag_id, EV_Flag *flag)

EV_ERR [EV_peek_flag](#) (EV_ID flag_id, EV_Flag *flag)

EV_ERR [EV_get_flag64](#) (EV_ID flag_id, EV_Flag64 *flag)

EV_ERR [EV_wait_flag64](#) (EV_ID flag_id, EV_Flag64 *flag)

EV_ERR [EV_peek_flag64](#) (EV_ID flag_id, EV_Flag64 *flag)
EV_ERR [EV_get_message](#) (EV_ID queue_id, EV_Message *message)
EV_ERR [EV_wait_message](#) (EV_ID queue_id, EV_Message *message)
EV_ERR [EV_peek_message](#) (EV_ID queue_id, EV_Message *message)
EV_ERR [EV_find_message_by_sender](#) (EV_ID queue_id, UINT32 senderInfo, EV_Message *message)
EV_ERR [EV_find_message_by_content](#) (EV_ID queue_id, UINT16 length, const void *compare_bytes,
EV_Message *message)
EV_ERR [EV_get_flag_fd](#) (EV_ID flag_id, int *fd)
EV_ERR [EV_get_queue_fd](#) (EV_ID queue_id, int *fd)
EV_ID [EV_moduleID_to_flagID](#) (UINT16 m_id)
EV_ID [EV_moduleID_to_flag64ID](#) (UINT16 m_id)
EV_ID [EV_moduleID_to_queueID](#) (UINT16 m_id)
EV_ERR [EV_create_flag_by_mID](#) (UINT16 m_id)
EV_ERR [EV_create_flag64_by_mID](#) (UINT16 m_id)
EV_ERR [EV_create_queue_by_mID](#) (UINT16 m_id, UINT8 length, UINT16 max_bytes,
EV_Message_Queue_Type type)
EV_ERR [EV_destroy_flag_by_mID](#) (UINT16 m_id)
EV_ERR [EV_destroy_queue_by_mID](#) (UINT16 m_id)
EV_ERR [EV_set_flag_by_mID](#) (UINT16 m_id, UINT32 bits)
EV_ERR [EV_set_flag64_by_mID](#) (UINT16 m_id, UINT64 bits)
EV_ERR [EV_send_message_by_mID](#) (UINT16 m_id, UINT16 bytes, const void *message, UINT32
senderInfo)

Detailed Description

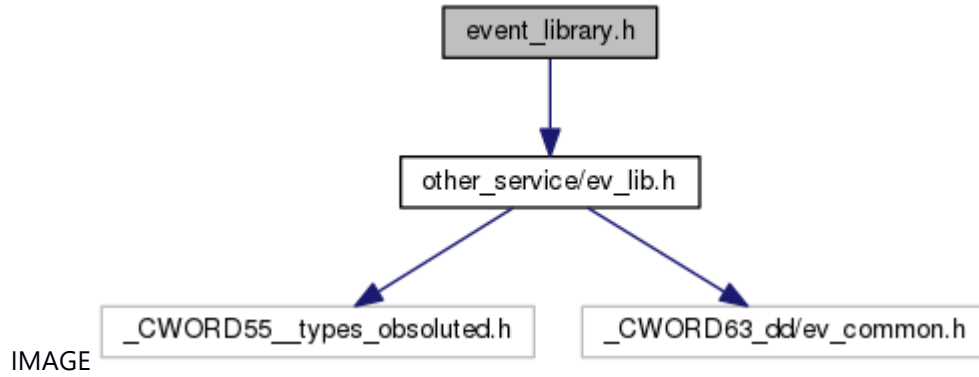
Event library API define head file.

event_library.h File Reference

include all event_library head files

```
#include <other_service/ev_lib.h>
```

Include dependency graph for event_library.h:



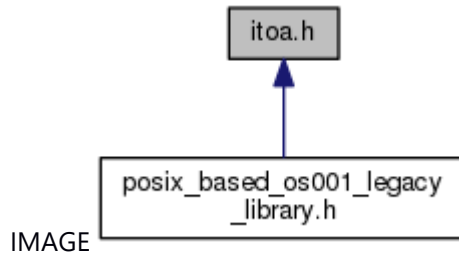
Detailed Description

include	all	event_library	head	files
---------	-----	---------------	------	-------

itoa.h File Reference

This file provides itoa API to convert integer value to null terminated string.

This graph shows which files directly or indirectly include this file:



Functions

char * [itoa](#) (int value, char buff[], int radix)

Detailed Description

This file provides itoa API to convert integer value to null terminated string.

posix_based_os001_legacy_library.h File Reference

include all posix_based_os001_legacy_library head files

```
#include <other_service/PosixBasedOS001ClockCycleApi.h>
```

```
#include <other_service/PosixBasedOS001TimeApi.h>
```

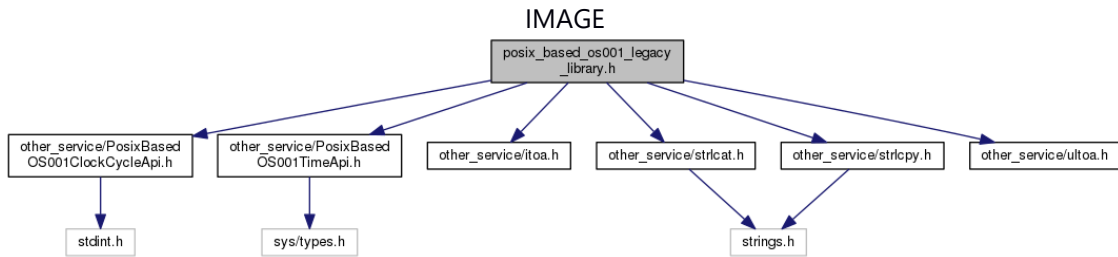
```
#include <other_service/itoa.h>
```

```
#include <other_service/strlcat.h>
```

```
#include <other_service/strlcpy.h>
```

```
#include <other_service/ultoa.h>
```

Include dependency graph for posix_based_os001_legacy_library.h:



Detailed Description

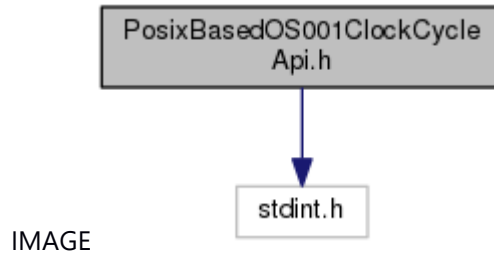
include all posix_based_os001_legacy_library head files

PosixBasedOS001ClockCycleApi.h File Reference

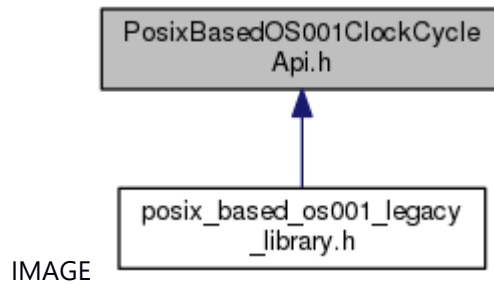
This file provides ClockCycle API to get number of clock cycles.

```
#include <stdint.h>
```

Include dependency graph for PosixBasedOS001ClockCycleApi.h:



This graph shows which files directly or indirectly include this file:



Functions

uint64_t [ClockCycle](#) (void)

Detailed Description

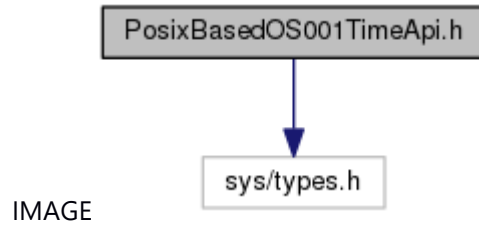
This file provides ClockCycle API to get number of clock cycles.

PosixBasedOS001TimeApi.h File Reference

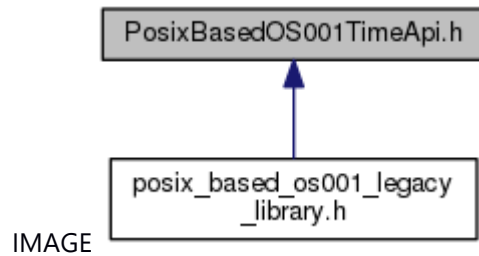
This file provides delay API to delay for the specified time.

```
#include <sys/types.h>
```

Include dependency graph for PosixBasedOS001TimeApi.h:



This graph shows which files directly or indirectly include this file:



Functions

unsigned int [delay](#) (unsigned int duration)

Detailed Description

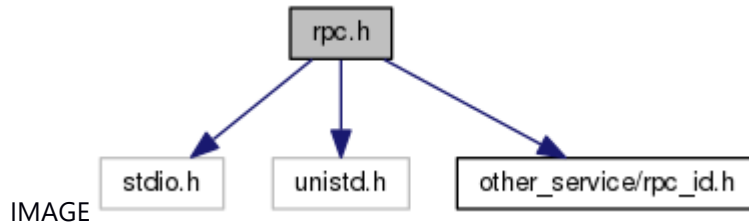
This file provides delay API to delay for the specified time.

rpc.h File Reference

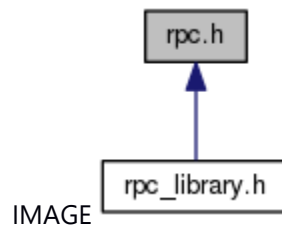
RPC library API define header file.

```
#include <stdio.h>
#include <unistd.h>
#include <other_service/rpc_id.h>
```

Include dependency graph for rpc.h:



This graph shows which files directly or indirectly include this file:



Classes

union [RPC_marshall_flag](#)

API call property stored struct. [More...](#)

struct [RPC_marshall_flag.bits](#)

Macros

```
#define OTHERSERVICE_RPC_H_
```

```
#define RPC\_MAX\_THREADS\_IN\_PROCESS 64
```

RPC_ID max number.

```
#define RPC\_MAX\_FD\_IN\_PROCESS 130
```

fd max number in one process. 0 = main/sub thread communication pipe fd. 1 = inotify() process listen fd. 2~129 = even num : API request data socket. odd num : secure stream socket.

```
#define RPC\_MAX\_API\_ARG\_NUM 8
```

API call parameter max num.

```
#define RPC\_MAX\_API\_ARG\_SIZE 1024
```

API call one parameter max byte.

```
#define RPC\_MAX\_API\_ARG\_TOTAL\_SIZE 1500
```

API call all parameter max byte.

```
#define RPC\_MAX\_APICALL\_QUEUE 16
    API call max queue num.
```

```
#define RPC\_NO\_PORT 0
    invalid port num
```

```
#define RPC\_NO\_ID RPC\_NO\_PORT
    The ID when destination is invalid.
```

```
#define RPC\_SELF\_ID RPC\_NO\_PORT
    The ID when destination is self.
```

```
#define RPC\_ANY\_ID RPC\_NO\_PORT
    The ID when destination is any.
```

```
#define RPC_DISPATCH_FUNC NULL
```

```
#define RPC\_START\_SERVER(ID) RPC_start((ID), RPC_DISPATCH_FUNC, NULL,
    NO\_SECURE\_CHECK)
```

```
#define RPC\_START\_SECURE\_SERVER(ID) RPC_start((ID), RPC_DISPATCH_FUNC, NULL,
    NEED\_SECURE\_CHECK)
```

```
#define RPC\_START\_CLIENT(pID) RPC_start((UINT16)RPC\_NO\_ID, NULL, (pID),
    NO\_SECURE\_CHECK)
    client start
```

```
#define NEED\_SECURE\_CHECK 1
    has secure check
```

```
#define NO\_SECURE\_CHECK 0
    has no secure check
```

```
#define REGISTERED 1
    registered
```

```
#define NO\_REGISTERED 0
    not registered
```

```
#define RPC_RETCODE_LEN 9
```

```
#define RPC_MARSHALL_FLAG_BITS_CODE 12
```

```
#define RPC_MARSHALL_FLAG_BITS_IS_VARARRAY 1
```

```
#define RPC_MARSHALL_FLAG_BITS_IS_POINTER 1
```

```
#define RPC_MARSHALL_FLAG_BITS_IN_OUT 2
```

```
#define rpc_malloc malloc
```

```
#define rpc_free free
```

Typedefs

```
typedef INT32 RPC\_Result
    RPC library function return type.
```

```
typedef UINT32 RPC\_ID
    RPC ID.
```

```
typedef RPC\_Result(* RPC_dispatch_func_t) (UINT16 api_num, const char *args_string, unsigned
    int args_size, char **ret_string, unsigned int *ret_bytes)
```

Enumerations

enum [rpc_result](#) { [RPC OK](#) = ***, [RPC ERR No Response](#) = ***, [RPC ERR Timeout](#),
[RPC ERR Busy](#), [RPC ERR API Error](#), [RPC ERR API Fatal](#), [RPC ERR Fatal](#),
[RPC ERR Configuration](#), [RPC ERR Server DeadLock](#), [RPC ERR Server Finish](#),
[RPC ERR Reject connect](#) }RPC library function return value.

Functions

[RPC Result RPC_start](#) ([RPC ID](#) id, RPC_dispatch_func_t func, [RPC ID](#) *pID, INT32 secure_check)
[RPC Result RPC_get fd](#) ([RPC ID](#) id, int *fd)
[RPC Result RPC_process API request](#) ([RPC ID](#) id)
void [RPC_end](#) ([RPC ID](#) id)
void [RPC_end all](#) (void)
[RPC Result RPC_set API timeout](#) (INT32 sec)
[RPC Result RPC_is server ready](#) ([RPC ID](#) id)
[RPC Result RPC_regist credential](#) (int uid_num, uid_t *uid_list, int gid_num, gid_t *gid_list)
[RPC Result RPC_get client credential](#) (uid_t *client_uid, gid_t *client_gid)
[RPC Result RPC_API_call](#) ([RPC ID](#) id, UINT16 api_num, const char *args_string, unsigned int
args_size, char **ret_string, unsigned int *ret_size)
int [RPC_demarshal arguments](#) (const char *from, unsigned int size, int need_alloc, int
num_args,...)
char * [RPC_marshall arguments](#) (unsigned int *size, int dont_marshall_out_args, int num_args,...)
void [RPC_marshall free](#) (int num,...)
void [RPC_free return string](#) (void *ptr)

Variables

char * [RPC_DEBUG](#)

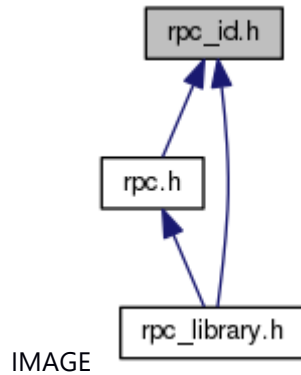
Detailed Description

RPC	library	API	define	header	file.
-----	---------	-----	--------	--------	-------

rpc_id.h File Reference

RPC library RPC_ID(port no) define.

This graph shows which files directly or indirectly include this file:



Macros

```
#define OTHERSERVICE\_RPCID\_H  
    define the RPC_ID used by program like #define XXX_RPC_ID ID but  
    RPC\_START\_CLIENT\(\) is not necessary  
  
#define test\_RPC\_ID 49999  
    sample(ID is used by RPCLibrary test program)  
  
#define MODEMANAGER_RPC_ID 51100 /* ModeManager */  
#define ACTIVITYMANAGER_RPC_ID 51101 /* ActivityManager */  
#define TIMERENTRYDRV_RPC_ID 53000 /* TimerEntryDrv */  
#define TSKM_RPC_ID 53001 /* TaskManager */  
#define MSGBRK_RPC_ID 53002 /* MessageBroker */  
#define IPMANAGER_RPC_ID 53003 /* IPManager */  
#define IPMANAGER_RPC_ID 53003 /* IPManager */  
#define DEVICEMANAGER_RPC_ID 53004 /* DeviceManager */  
#define _CWORD87_LAN_RPC_ID 53005 /* Communication(_CWORD87_LAN) */  
#define CAN_RPC_ID 53006 /* Communication(CAN) */  
#define _CWORD83_RPC_ID 53007 /* Communication(_CWORD83_) */  
#define SENSOR_RPC_ID 53008 /* Vehicle(Sensor) */  
#define GPS_RPC_ID 53009 /* Vehicle(GPS) */  
#define RESMGR_RPC_ID 53010 /* ResourceManager */  
#define GRAPHICS_RPC_ID 53011 /* Graphic */
```

Detailed Description

RPC	library	RPC_ID(port	no)	define.
-----	---------	-------------	-----	---------

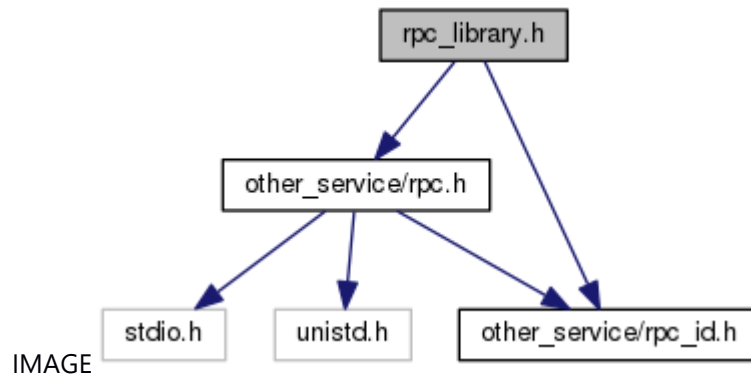
rpc_library.h File Reference

include all rpc_library head files

```
#include <other_service/rpc.h>
```

```
#include <other_service/rpc_id.h>
```

Include dependency graph for rpc_library.h:



Detailed Description

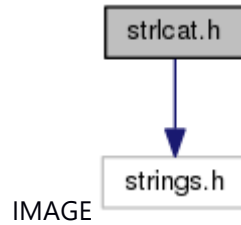
include	all	rpc_library	head	files
---------	-----	-------------	------	-------

strlcat.h File Reference

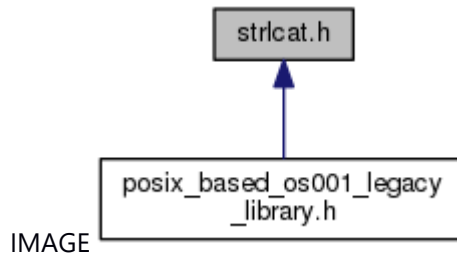
This file provides strlcat API to concatenate strings.

```
#include <strings.h>
```

Include dependency graph for strlcat.h:



This graph shows which files directly or indirectly include this file:



Functions

size_t [strlcat](#) (char *dst, const char *src, size_t siz)

Detailed Description

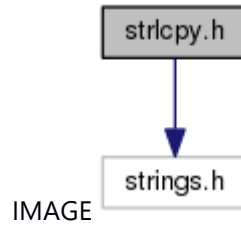
This file provides strlcat API to concatenate strings.

strncpy.h File Reference

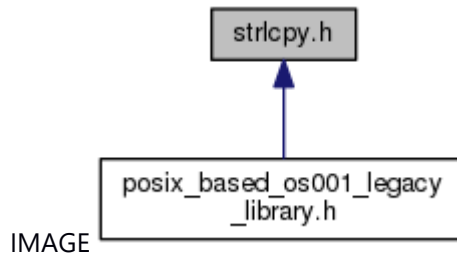
This file provides API to copy string.

```
#include <strings.h>
```

Include dependency graph for strncpy.h:



This graph shows which files directly or indirectly include this file:



Functions

size_t [strncpy](#) (char *dst, const char *src, size_t siz)

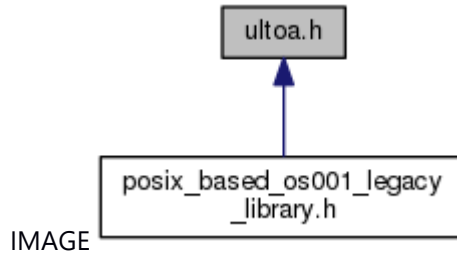
Detailed Description

This file provides API to copy string.

ultoa.h File Reference

This file provides ultoa API to convert unsigned long type to NULL terminated character string.

This graph shows which files directly or indirectly include this file:



Functions

char * [ultoa](#) (unsigned long value, char *buf, int radix)

Detailed Description

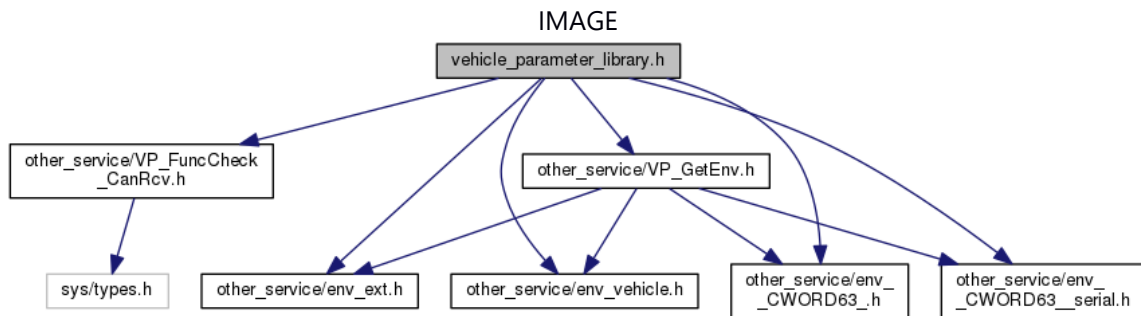
This file provides ultoa API to convert unsigned long type to NULL terminated character string.

vehicle_parameter_library.h File Reference

include all vehicle_parameter_library head files

```
#include <other_service/VP_FuncCheck_CanRcv.h>  
#include <other_service/VP_GetEnv.h>  
#include <other_service/env_CWORD63_.h>  
#include <other_service/env_CWORD63_serial.h>  
#include <other_service/env_vehicle.h>  
#include <other_service/env_ext.h>
```

Include dependency graph for vehicle_parameter_library.h:



Detailed Description

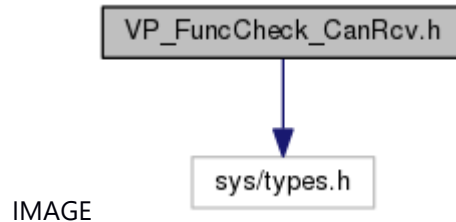
include all vehicle_parameter_library head files

VP_FuncCheck_CanRcv.h File Reference

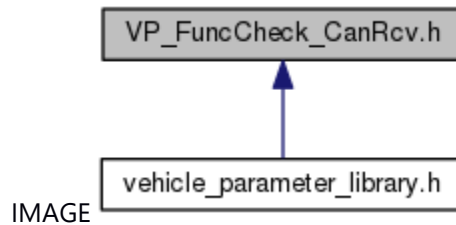
Get function existence API header file.

```
#include <sys/types.h>
```

Include dependency graph for VP_FuncCheck_CanRcv.h:



This graph shows which files directly or indirectly include this file:



Macros

```
#define XM\_AUDIO ((u_int8_t)1)
#define CWORD76 ((u_int8_t)2)
#define VP\_CANRCV\_CWORD27 ((u_int8_t)3)
#define WIFI\_HOTSPOT ((u_int8_t)4)
#define NA\_STD\_TRAF\_LIMIT ((u_int8_t)5)
#define HD\_DATA ((u_int8_t)6)
#define FUNCTION\_CHECK\_RESULT\_OK ((u_int8_t)0)
#define FUNCTION\_CHECK\_RESULT\_NG ((u_int8_t)1)
#define FUNCTION\_CHECK\_RESULT\_UNKNOWN ((u_int8_t)2)
#define FUNCTION\_CHECK\_RESULT\_OUT\_OF\_NA ((u_int8_t)3)
```

Functions

u_int8_t [VP_FuncCheck_CanRcv](#) (u_int8_t FunctionName)

Detailed Description

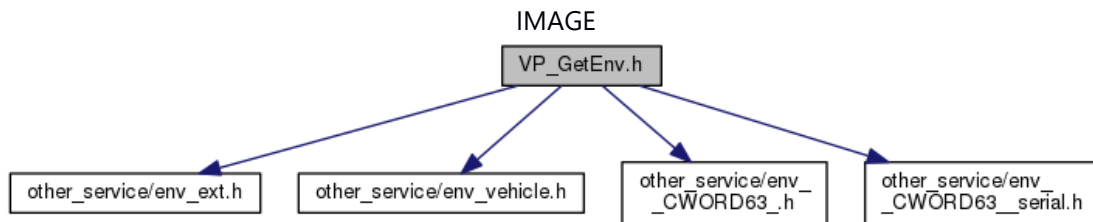
Get function existence API header file.

VP_GetEnv.h File Reference

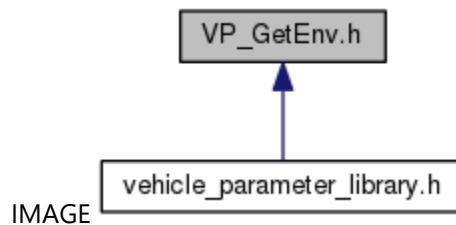
Get vehicle parameter environment variable API header file.

```
#include <other_service/env_ext.h>
#include <other_service/env_vehicle.h>
#include <other_service/env_CWORD63.h>
#include <other_service/env_CWORD63_serial.h>
```

Include dependency graph for VP_GetEnv.h:



This graph shows which files directly or indirectly include this file:



Macros

```
#define VP\_MAX\_LENGTH 128
#define CWORD63\_AREA "_CWORD63_AREA"
#define CWORD63\_GRADE "_CWORD63_GRADE"
#define CWORD63\_UNIT "_CWORD63_UNIT"
#define CWORD63\_BRAND "_CWORD63_BRAND"
#define CWORD63\_HAS\_CWORD80 "_CWORD63_HAS_CWORD80_"
#define CWORD63\_BOARD "_CWORD63_BOARD"
#define CWORD63\_CWORD86 "_CWORD63_CWORD86_"
```

Functions

```
void VP\_GetEnv (const char *pEnvStr, char *pEnvBuff)
```

Detailed Description

Get vehicle parameter environment variable API header file.