

libcom3.h

[Go to the documentation of this file.](#)

```
1Â #pragma once      2Â      3Â #if defined __GNUC__ && defined VFI_COM_SHARED
<sys/types.h>      10Â #include <ipc/jsobject.h>      11Â #include <exception>
12Â      13Â namespace vficom {      14Â      15Â namespace cmd {      16Â      17Â COM
<! String, can be empty      210Â COM3_DllSpec extern const char bssid[];      211
Â COM3_DllSpec extern const char adhocalpha[];      212Â COM3_DllSpec extern const
char pairw[];      213Â COM3_DllSpec extern const char group[];      214Â
COM3_DllSpec extern const char km[];      215Â COM3_DllSpec extern const
char perc[];      216Â COM3_DllSpec extern const char dbm[];      217Â
COM3_DllSpec extern const char freq[];      218Â COM3_DllSpec extern const
char proto[];      219Â COM3_DllSpec extern const char std[];      220Â      221
Â } // namespace json_keys      222Â      223Â } // namespace wlan_scan      224Â
225Â } // namespace out      226Â      227Â } // namespace cmdparam      228Â      229
Â namespace event {      230Â enum EventType {      231Â      EVT_URC_STK = 0,      232
Â      EVT_URC_USSD = 1,      233Â      EVT_MAX_ENUM = 0x7FFFFFFF      234Â };      235
Â      236Â namespace stk {      237Â enum STKURCProCommand {      238Â
SET_UP_EVENT_LIST      = 5,      239Â      SELECT_ITEM      = 36,      240Â
MENU_SELECTION      = 37,      241Â      STK_URC_PRO_MAX_ENUM = 0x7FFFFFFF      242
Â };      243Â      244Â COM3_DllSpec extern const char urc[];
245Â COM3_DllSpec extern const char proactive_command[];      246Â
COM3_DllSpec extern const char parameters[];      247Â COM3_DllSpec
extern const char timeout[];      248Â      249Â namespace
set_up_event_list {      250Â      251Â COM3_DllSpec extern const char event_list
[];      252Â      253Â } // namespace set_up_event_list      254Â      255Â namespace
select_item {      256Â      257Â COM3_DllSpec extern const char cmd_qualifier
[];      258Â COM3_DllSpec extern const char alpha[];      259Â
COM3_DllSpec extern const char item_id[];      260Â COM3_DllSpec
extern const char total_items[];      261Â COM3_DllSpec extern const
char item_text[];      262Â COM3_DllSpec extern const char next_action
[];      263Â COM3_DllSpec extern const char default_item[];      264
Â COM3_DllSpec extern const char icon_id[];      265Â COM3_DllSpec
extern const char icon_id_list_elem[];      266Â      267Â } // namespace
select_item      268Â      269Â namespace menu_selection {      270Â      271Â
COM3_DllSpec extern const char cmd_qualifier[];      272Â COM3_DllSpec
extern const char alpha[];      273Â COM3_DllSpec extern const
char item_id[];      274Â COM3_DllSpec extern const char total_items
[];      275Â COM3_DllSpec extern const char item_text[];      276
Â COM3_DllSpec extern const char next_action[];      277Â COM3_DllSpec
extern const char icon_id[];      278Â COM3_DllSpec extern const
char icon_id_list_elem[];      279Â      280Â } // namespace menu_selection      281
Â      282Â } // namespace stk      283Â      284Â namespace ussd {      285Â      286
Â // Names correspond to 3GPP TS 27.00, Chapter 7.15, Unstructured
```

```

supplementary service data +CUSD 287Â COM3_DllSpec extern const char m
[]; 288Â COM3_DllSpec extern const char str[]; 289Â COM3_DllSpec
extern const char dcs[]; 290Â 291Â } // namespace ussd 292Â 293
Â } // namespace event 294Â 295Â namespace status { 296Â 297Â enum
PipeStatus { 298Â OK = 0, 299Â INVALID = 1,
300Â NOSUPPORT = 2, 301Â NOTREADY = 3, 302Â
FAILED = 5, 303Â TIMEOUT = 6, 304Â STARVATION
= 7, 305Â USER_INTERRUPT = 8, 306Â MAX_ENUM =
0x7FFFFFFF 307Â }; 308Â 309Â } // namespace status 310Â 315Â class
COM3_DllSpec Exception : std::exception { 316Â char description[64
]; 317Â status::PipeStatus error; 318Â 319Â public: 320Â
Exception(enum status::PipeStatus error, const char *description); 321
Â 322Â virtual const char *what() const throw(); 323Â
status::PipeStatus getError() const; 324Â }; 325Â 326Â class
COM3_DllSpec Milliseconds { 327Â public: 328Â u_int64_t millis; 329
Â 330Â Milliseconds(u_int32_t millis); 331Â ~Milliseconds();
332Â }; 333Â 334Â class COM3_DllSpec CommandPipe { 335Â class
ComdaemonPipe *impl; 336Â 337Â // Copy not valid 338Â
CommandPipe(const CommandPipe &commandPipe); 339Â void operator=(const
CommandPipe &commandPipe); 340Â 341Â public: 342Â CommandPipe
(); 343Â virtual ~CommandPipe(); 344Â 354Â enum
status::PipeStatus sendRecv(const vfiipc::JSObject &in, vfiipc::JSObject &
out, Milliseconds millis); 355Â 367Â enum status::PipeStatus
sendRecv(const vfiipc::JSObject &in, vfiipc::JSObject &out, Milliseconds
millis, int interruptFD); 368Â }; 369Â 370Â class COM3_DllSpec
EventPipe { 371Â class ComdaemonPipe *impl; 372Â 373Â public: 374
Â EventPipe(event::EventType type); 375Â ~EventPipe(); 376Â
377Â // Copy not valid 378Â EventPipe(const EventPipe &eventPipe
); 379Â void operator=(const EventPipe &eventPipe); 380Â 389Â
enum status::PipeStatus recv(vfiipc::JSObject &out, Milliseconds millis);
390Â 401Â enum status::PipeStatus recv(vfiipc::JSObject &out,
Milliseconds millis, int interruptFD); 402Â }; 403Â 404Â } //
namespace vfiicom 405Â 406Â #undef COM3_DllSpec

```