

madk_pp_protocol.h

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

```
1
2 /*****
3  * Company:      Verifone      3  * Author:        GSS R&D Germany      4
4  * Content:      SDI-Server    5
5  *****/
6 #ifndef _MADK_PP_PROTOCOL_H_      7 #define _MADK_PP_PROTOCOL_H_      8
9 #include <vector>      10 #include <string>      11
12 #include <pthread.h>      14 struct mADK_PP_Prot_Loader;      16
17 struct ProtStatus;      18 struct ConnectInfo;      19
20 struct UdsInfo;      23 class mADK_PP_Prot      24 {      25
26     public:      27         enum ProtocolState      28     {      29
30         PS_Stopped = 0,      31         PS_Running = 1,
32         PS_Aborted = 2      33     };      34         enum AbortState
35     {      36         AS_NotAborted = 0,      37
38         AS_AbortIdle = 1,      39         AS_ConnectionLost = 2      41
40     };      42         43     private:      44         pthread_t proto_thread;      45
46         pthread_mutex_t mutex;      47         pthread_cond_t cond;
48         std::vector<unsigned char> cmdbuf;      49         char prot_type;      50
51         unsigned msg_id;      52         unsigned msg_id_payment;
53         unsigned msg_id_nested;      54         enum ProtocolState
55         prot_state;      56         bool cmd_active;      57         bool nested_cmd;
58     {      59         Cmd_Any = 0,      60
61         Cmd_Abort = 1,      62
63         Cmd_LockPayment = 2,      64
65         Cmd_ResetLink = 3,      66
67         Cmd_Check4Update = 4,      68
69         Cmd_ExtButton = 5,      70
71         Cmd_EmvCbResponse = 6,      71
73         Cmd_CtrlCbResponse = 7,      72
75         Cmd_TrgrCbResponse = 8,      73
77         Cmd_PaymentNotificationCb = 9      74     };      75         enum
76         RequestType      77     {      78         Request_None = 0,      79
80         Request_EMV = 1,      81         Request_Ctrl = 2,      82
83         Request_Trgr = 3      84     };      85
86         RequestType req_type;      87         bool recovery_mode;      88         bool
89         busy;      90         bool cancelled;      91         bool
92         unlock_payment_pending;      93         void *main_handle;      94
95         void *busy_handle;      96         static mADK_PP_Prot *obj;
97         // singleton      98         // disable calling copy constructor and assignment
99     }
```