

Pegoda RD710/RD852 Reader Firmware
2.0.0

Generated by Doxygen 1.7.3

Tue Jun 12 2012 10:27:23

Contents

1	Module Index	1
1.1	Modules	1
2	Data Structure Index	3
2.1	Data Structures	3
3	File Index	5
3.1	File List	5
4	Module Documentation	7
4.1	APDUs Defines	7
4.1.1	Define Documentation	8
4.1.1.1	P3FW_APDU_CC_EXT_INS	8
4.1.1.2	P3FW_APDU_CC_EXT_MANAGE_SESSION	8
4.1.1.3	P3FW_APDU_CC_EXT_SWITCH_PROTOCOL	8
4.1.1.4	P3FW_APDU_CC_EXT_TRANS_EXCHANGE	8
4.1.1.5	P3FW_APDU_CLASS	8
4.1.1.6	P3FW_APDU_G_AUTH_CMD_INS	8
4.1.1.7	P3FW_APDU_GET_DATA_INS	8
4.1.1.8	P3FW_APDU_INS	8
4.1.1.9	P3FW_APDU_Lc	8
4.1.1.10	P3FW_APDU_Le	9
4.1.1.11	P3FW_APDU_LOAD_KEY_INS	9
4.1.1.12	P3FW_APDU_P1	9
4.1.1.13	P3FW_APDU_P2	9
4.1.1.14	P3FW_APDU_PAYLOAD	9
4.1.1.15	P3FW_APDU_READ_BIN	9
4.1.1.16	P3FW_APDU_UPDATE_BIN	9
4.2	CCID functions	9
4.2.1	Detailed Description	11
4.2.2	Define Documentation	11
4.2.2.1	P3FW_CCID_BULK_HEADER	11
4.2.2.2	P3FW_CCID_BULK_OUT_REQ_ESCAPE	12
4.2.2.3	P3FW_CCID_BULK_OUT_REQ_GETPARAMETERS	12
4.2.2.4	P3FW_CCID_BULK_OUT_REQ_GETSLOTSTATUS	12
4.2.2.5	P3FW_CCID_BULK_OUT_REQ_ICCPOWEROFF	12
4.2.2.6	P3FW_CCID_BULK_OUT_REQ_ICCPOWERON	12
4.2.2.7	P3FW_CCID_BULK_OUT_REQ_SETPARAMETERS	12
4.2.2.8	P3FW_CCID_BULK_OUT_REQ_XFRBLOCK	12

4.2.2.9	P3FW_CCID_CARD_IN_SLOT	12
4.2.2.10	P3FW_CCID_CARD_OUT_SLOT	13
4.2.2.11	P3FW_CCID_ERROR_SLOT_BUSY	13
4.2.2.12	P3FW_CCID_ERROR_SLOT_CMD_NOT_SUPPORTED	13
4.2.2.13	P3FW_CCID_ERROR_SLOT_HW_ERROR	13
4.2.2.14	P3FW_CCID_ERROR_SLOT_ICC_MUTE	13
4.2.2.15	P3FW_CCID_ERROR_SLOT_NOT_EXIST	13
4.2.2.16	P3FW_CCID_ERROR_SLOT_XFR_OVERRUN	13
4.2.2.17	P3FW_CCID_HEADER_LENGTH_BYTE_1	13
4.2.2.18	P3FW_CCID_HEADER_LENGTH_BYTE_2	14
4.2.2.19	P3FW_CCID_HEADER_LENGTH_BYTE_3	14
4.2.2.20	P3FW_CCID_HEADER_LENGTH_BYTE_4	14
4.2.2.21	P3FW_CCID_HEADER_MESSAGE_TYPE	14
4.2.2.22	P3FW_CCID_HEADER_MSG_BYTE_1	14
4.2.2.23	P3FW_CCID_HEADER_MSG_BYTE_2	14
4.2.2.24	P3FW_CCID_HEADER_MSG_BYTE_3	14
4.2.2.25	P3FW_CCID_HEADER_SEQ	14
4.2.2.26	P3FW_CCID_HEADER_SLOT	15
4.2.2.27	P3FW_CCID_INT_IN_NOTIFY_SLOT_CHANGE	15
4.2.2.28	P3FW_CCID_RDR_TO_PC_ESCAPE	15
4.2.2.29	P3FW_CCID_RDR_TO_PC_PARAMETERS	15
4.2.2.30	P3FW_CCID_RDR_TO_PC_SLOT_DATA_BLOCK	15
4.2.2.31	P3FW_CCID_RDR_TO_PC_SLOT_STATUS	15
4.2.2.32	P3FW_CCID_STATUS_CMD_FAILED	15
4.2.3	Function Documentation	15
4.2.3.1	p3fw_ccid_bh_escape	15
4.2.3.2	p3fw_ccid_bh_get_parameters	16
4.2.3.3	p3fw_ccid_bh_get_slot_status	16
4.2.3.4	p3fw_ccid_bh_icc_power_off	16
4.2.3.5	p3fw_ccid_bh_icc_power_on	16
4.2.3.6	p3fw_ccid_bh_set_parameters	17
4.2.3.7	p3fw_ccid_bh_xfer	17
4.2.3.8	p3fw_ccid_get_input_payload_buffer	17
4.2.3.9	p3fw_ccid_get_input_payload_length	17
4.2.3.10	p3fw_ccid_get_output_payload_buffer	17
4.2.3.11	p3fw_ccid_input_header_get_byte	18
4.2.3.12	p3fw_ccid_send_frame	18
4.2.3.13	p3fw_ccid_send_notify	18
4.2.3.14	p3fw_ccid_set_output_payload_length	18
4.2.3.15	p3fw_ccid_th_dispatch	18
4.3	Mode Functions	19
4.3.1	Define Documentation	22
4.3.1.1	P3FW_DM_CHECK_LENGTH_EQUAL	22
4.3.1.2	P3FW_DM_CHECK_LENGTH_LARGER	22
4.3.1.3	P3FW_DM_CHECK_STATUS	22
4.3.1.4	P3FW_DM_CID	23
4.3.1.5	P3FW_DM_CID_FREE	23
4.3.1.6	P3FW_DM_CID_GET_FREE	23
4.3.1.7	P3FW_DM_CID_INIT	23
4.3.1.8	P3FW_DM_CONTACT_CARD	23

4.3.1.9	P3FW_DM_CONTACTCARD_ACTIVATE_CARD	23
4.3.1.10	P3FW_DM_CONTACTCARD_CLOCK_START	23
4.3.1.11	P3FW_DM_CONTACTCARD_CLOCK_STOP	23
4.3.1.12	P3FW_DM_CONTACTCARD_COLD_RESET	24
4.3.1.13	P3FW_DM_CONTACTCARD_DEACTIVATE_CARD	24
4.3.1.14	P3FW_DM_CONTACTCARD_PPS	24
4.3.1.15	P3FW_DM_CONTACTCARD_PRESENCE_CHECK	24
4.3.1.16	P3FW_DM_CONTACTCARD_TRANSMIT_DATA	24
4.3.1.17	P3FW_DM_CONTACTCARD_WARM_RESET	24
4.3.1.18	P3FW_DM_HAL	24
4.3.1.19	P3FW_DM_HAL_APP_PROT_SET	24
4.3.1.20	P3FW_DM_HAL_EXEC_CMD	25
4.3.1.21	P3FW_DM_HAL_GET_CFG	25
4.3.1.22	P3FW_DM_HAL_INIT	25
4.3.1.23	P3FW_DM_HAL_MFC_AUTH	25
4.3.1.24	P3FW_DM_HAL_MFC_AUTH_KEY	25
4.3.1.25	P3FW_DM_HAL_SET_CFG	25
4.3.1.26	P3FW_DM_HAL_WAIT	25
4.3.1.27	P3FW_DM_HAL_XCHG	25
4.3.1.28	P3FW_DM_KSTOR	26
4.3.1.29	P3FW_DM_KSTOR_CHG_KUC	26
4.3.1.30	P3FW_DM_KSTOR_FORMAT_KEY	26
4.3.1.31	P3FW_DM_KSTOR_GET_CFG_STR	26
4.3.1.32	P3FW_DM_KSTOR_GET_CONFIG	26
4.3.1.33	P3FW_DM_KSTOR_GET_KEY	26
4.3.1.34	P3FW_DM_KSTOR_GET_KEY_ENTRY	26
4.3.1.35	P3FW_DM_KSTOR_GET_KUC	26
4.3.1.36	P3FW_DM_KSTOR_INIT	27
4.3.1.37	P3FW_DM_KSTOR_SET_CFG_STR	27
4.3.1.38	P3FW_DM_KSTOR_SET_CONFIG	27
4.3.1.39	P3FW_DM_KSTOR_SET_FULL_KEY	27
4.3.1.40	P3FW_DM_KSTOR_SET_KEY	27
4.3.1.41	P3FW_DM_KSTOR_SET_KEY_POS	27
4.3.1.42	P3FW_DM_KSTOR_SET_KUC	27
4.3.1.43	P3FW_DM_L3	27
4.3.1.44	P3FW_DM_L3_ACT_CARD	28
4.3.1.45	P3FW_DM_L3_ANTICOL	28
4.3.1.46	P3FW_DM_L3_GET_SER	28
4.3.1.47	P3FW_DM_L3_HLTA	28
4.3.1.48	P3FW_DM_L3_INIT	28
4.3.1.49	P3FW_DM_L3_REQA	28
4.3.1.50	P3FW_DM_L3_SELECT	28
4.3.1.51	P3FW_DM_L3_WKUA	28
4.3.1.52	P3FW_DM_L3_XCHG	29
4.3.1.53	P3FW_DM_L4	29
4.3.1.54	P3FW_DM_L4_DESELECT	29
4.3.1.55	P3FW_DM_L4_GET_CFG	29
4.3.1.56	P3FW_DM_L4_INIT	29
4.3.1.57	P3FW_DM_L4_PRES_CHECK	29
4.3.1.58	P3FW_DM_L4_RESET_PROTO	29

4.3.1.59	P3FW_DM_L4_SET_CFG	29
4.3.1.60	P3FW_DM_L4_SET_PROTO	30
4.3.1.61	P3FW_DM_L4_XCHG	30
4.3.1.62	P3FW_DM_L4A	30
4.3.1.63	P3FW_DM_L4A_ACT_CARD	30
4.3.1.64	P3FW_DM_L4A_GET_PROTO_PARM	30
4.3.1.65	P3FW_DM_L4A_INIT	30
4.3.1.66	P3FW_DM_L4A_PPS	30
4.3.1.67	P3FW_DM_L4A_RATS	30
4.3.1.68	P3FW_DM_RO	31
4.3.1.69	P3FW_DM_RO_CONF_OVER	31
4.3.1.70	P3FW_DM_RO_FIELD_OFF	31
4.3.1.71	P3FW_DM_RO_FIELD_ON	31
4.3.1.72	P3FW_DM_RO_FIELD_RESET	31
4.3.1.73	P3FW_DM_RO_GET_CONF	31
4.3.1.74	P3FW_DM_RO_GET_STATUS	31
4.3.1.75	P3FW_DM_RO_LEDS_OFF	31
4.3.1.76	P3FW_DM_RO_LEDS_ON	32
4.3.1.77	P3FW_DM_RO_READ_REG	32
4.3.1.78	P3FW_DM_RO_RESET	32
4.3.1.79	P3FW_DM_RO_SET_CONF	32
4.3.1.80	P3FW_DM_RO_SET_PCSC_MODE	32
4.3.1.81	P3FW_DM_RO_TEST_MODE	32
4.3.1.82	P3FW_DM_RO_WRITE_REG	32
4.3.1.83	P3FW_DM_XCHG	32
4.3.1.84	P3FW_DM_XCHG_INIT	33
4.3.1.85	P3FW_DM_XCHG_L3	33
4.3.1.86	P3FW_DM_XCHG_L4	33
4.3.1.87	P3FW_DM_XCHG_MFC_AUTH	33
4.3.1.88	P3FW_DM_XCHG_MFC_AUTH_KEY	33
4.3.1.89	P3FW_DM_XCHG_PC	33
4.3.1.90	P3FW_DM_XCHG_RAW	33
4.3.2	Function Documentation	33
4.3.2.1	p3fw_dm_cid	33
4.3.2.2	p3fw_dm_contact_card	34
4.3.2.3	p3fw_dm_get_class	34
4.3.2.4	p3fw_dm_get_input_payload	34
4.3.2.5	p3fw_dm_get_input_payload_length	34
4.3.2.6	p3fw_dm_get_input_pointer_uint8_t	34
4.3.2.7	p3fw_dm_get_input_uint16_t	34
4.3.2.8	p3fw_dm_get_input_uint32_t	34
4.3.2.9	p3fw_dm_get_input_uint8_t	35
4.3.2.10	p3fw_dm_get_instruction	35
4.3.2.11	p3fw_dm_get_output_payload	35
4.3.2.12	p3fw_dm_get_output_pointer_uint8_t	35
4.3.2.13	p3fw_dm_get_slot_index	35
4.3.2.14	p3fw_dm_hal	35
4.3.2.15	p3fw_dm_keystore	35
4.3.2.16	p3fw_dm_l3	36
4.3.2.17	p3fw_dm_l4	36

4.3.2.18	p3fw_dm_l4a	36
4.3.2.19	p3fw_dm_ro	36
4.3.2.20	p3fw_dm_send_frame	36
4.3.2.21	p3fw_dm_xchg	36
4.3.2.22	p3fw_dm_xfer	36
4.4	Firmware	37
4.4.1	Define Documentation	40
4.4.1.1	P3FW_BUILD	40
4.4.1.2	P3FW_CFG_BOOTLOADER_ACTIVE	40
4.4.1.3	P3FW_CFG_BOOTLOADER_VERSION	41
4.4.1.4	P3FW_CFG_CONT_TIMING	41
4.4.1.5	P3FW_CFG_GET_CONT_TIMING	41
4.4.1.6	P3FW_CFG_MAX_SLOTS	41
4.4.1.7	P3FW_CFG_SET_DIP_SWITCHES	41
4.4.1.8	P3FW_CHECK_STATUS	41
4.4.1.9	P3FW_CONTACT_SLOTS_MASK	42
4.4.1.10	P3FW_ERR_AND_NFO_LOOP_BFL_ERROR	42
4.4.1.11	P3FW_ERR_AND_NFO_LOOP_CHIP_NOT_DETECTED	42
4.4.1.12	P3FW_ERR_AND_NFO_LOOP_COM_ERROR	42
4.4.1.13	P3FW_ERR_AND_NFO_LOOP_DONE	42
4.4.1.14	P3FW_ERR_AND_NFO_LOOP_ERASE_FAILD	42
4.4.1.15	P3FW_ERR_AND_NFO_LOOP_FLASH_FAILD	42
4.4.1.16	P3FW_ERR_AND_NFO_LOOP_HAL_CAN_NOT_BE_SET	42
4.4.1.17	P3FW_ERR_AND_NFO_LOOP_HW_ERROR	43
4.4.1.18	P3FW_ERR_AND_NFO_LOOP_OS_ERROR	43
4.4.1.19	P3FW_ERR_AND_NFO_LOOP_UNKNOWN_ERROR	43
4.4.1.20	P3FW_ERR_AND_NFO_LOOP_USB_ERROR	43
4.4.1.21	P3FW_KEYSTORE_NUM_KEYS	43
4.4.1.22	P3FW_KEYSTORE_NUM_VERS	43
4.4.1.23	P3FW_MAJOR	43
4.4.1.24	P3FW_MAX_ATQB_LEN	43
4.4.1.25	P3FW_MAX_ATS_LEN	44
4.4.1.26	P3FW_MAX_CCID_BUFFER_LEN	44
4.4.1.27	P3FW_MAX_CONTACT_SLOTS	44
4.4.1.28	P3FW_MAX_CONTACTLESS_SLOTS	44
4.4.1.29	P3FW_MAX_HAL_RX_BUFFER	44
4.4.1.30	P3FW_MAX_HAL_TX_BUFFER	44
4.4.1.31	P3FW_MAX_KEY_LEN	44
4.4.1.32	P3FW_MAX_UID_A_LEN	44
4.4.1.33	P3FW_MAX_UID_B_LEN	44
4.4.1.34	P3FW_MINOR	45
4.4.1.35	P3FW_PCSC_FSDI	45
4.4.1.36	P3FW_PCSC_PROTO_PARAMS_LEN_T0	45
4.4.1.37	P3FW_PCSC_PROTO_PARAMS_LEN_T1	45
4.4.1.38	P3FW_USB_BULK_IN_EP	45
4.4.1.39	P3FW_USB_BULK_OUT_EP	45
4.4.1.40	P3FW_USB_INT_IN_EP	45
4.4.2	Typedef Documentation	45
4.4.2.1	p3fw_cc_slot	45

4.4.2.2	p3fw_cl_slot	45
4.4.3	Enumeration Type Documentation	46
4.4.3.1	p3fw_bal_configuration	46
4.4.3.2	p3fw_external_interface	46
4.4.3.3	p3fw_ic	46
4.4.3.4	p3fw_mode	46
4.4.3.5	p3fw_pcsc_mode	47
4.4.3.6	p3fw_pcsc_protocol	47
4.4.3.7	p3fw_sam	47
4.4.3.8	p3fw_slot_types	47
4.4.4	Function Documentation	48
4.4.4.1	p3fw_ext_if_init_usb	48
4.4.4.2	p3fw_flash_erase_config	48
4.4.4.3	p3fw_flash_get_config	48
4.4.4.4	p3fw_flash_read_serial	49
4.4.4.5	p3fw_flash_set_config	49
4.4.4.6	p3fw_invoke_err_and_nfo_mode	49
4.4.4.7	p3fw_slot_add_l3a_card	50
4.4.4.8	p3fw_slot_add_l3b_card	50
4.4.4.9	p3fw_slot_add_l4_card	50
4.4.4.10	p3fw_slot_get_atr	50
4.4.4.11	p3fw_slot_remove_cl_card	51
4.4.4.12	p3fw_slot_reset_all_slots	51
4.4.4.13	p3fw_slots_init	51
4.4.4.14	p3fw_task_ccid_execute	52
4.4.4.15	p3fw_task_demo_mode	52
4.4.4.16	p3fw_task_poll_and_activate	52
4.4.4.17	p3fw_timing_init	52
4.4.4.18	p3fw_timing_start	53
4.4.4.19	p3fw_timing_stop	53
4.4.5	Variable Documentation	53
4.4.5.1	p3fw	53
4.5	Initialization	53
4.5.1	Define Documentation	54
4.5.1.1	P3FW_HW_SIGNAL_ANTENNA	54
4.5.1.2	P3FW_HW_SIGNAL_BEEPER	54
4.5.1.3	P3FW_HW_SIGNAL_YELLOW_2	54
4.5.1.4	P3FW_HW_SIGNAL_YELLOW_3	54
4.5.1.5	P3FW_HW_SIGNAL_YELLOW_4	54
4.5.2	Function Documentation	54
4.5.2.1	p3fw_hw_ctrl_signal	54
4.5.2.2	p3fw_hw_init	54
4.6	PCSCs	55
4.6.1	Function Documentation	55
4.6.1.1	p3fw_pcsc_send_apdu	55
4.6.1.2	p3fw_pcsc_std_ext	56
4.6.1.3	p3fw_pcsc_std_ext_do_auth	56
4.6.1.4	p3fw_pcsc_std_ext_prepare_l3_card	57

5 Data Structure Documentation

59

5.1	_p3fw_cc_slot Struct Reference	59
5.1.1	Detailed Description	59
5.1.2	Field Documentation	59
5.1.2.1	bPPP	59
5.1.2.2	eProtocolType	59
5.1.2.3	eSlotType	60
5.2	_p3fw_cl_slot Struct Reference	60
5.2.1	Detailed Description	60
5.2.2	Field Documentation	60
5.2.2.1	bAts	60
5.2.2.2	bBlockPoll	61
5.2.2.3	bPPP	61
5.2.2.4	bSak	61
5.2.2.5	bUid	61
5.2.2.6	bUidLength	61
5.2.2.7	eProtocolType	61
5.2.2.8	eSlotType	61
5.2.2.9	pHal	61
5.2.2.10	sISO14443L3a	62
5.2.2.11	sISO14443L3b	62
5.2.2.12	sISO14443L4	62
5.2.2.13	sISO14443L4a	62
5.2.2.14	sMfc	62
5.2.2.15	sMful	62
5.2.2.16	sMifare	62
5.3	cpot_atr_frame Struct Reference	63
5.3.1	Detailed Description	63
5.3.2	Field Documentation	63
5.3.2.1	historicalC	63
5.3.2.2	state	63
5.3.2.3	TAi	63
5.3.2.4	TBi	63
5.3.2.5	TCi	63
5.3.2.6	TCK	63
5.3.2.7	TDi	64
5.4	p2_fw_SAM_ctrl_ Struct Reference	64
5.4.1	Detailed Description	64
5.4.2	Field Documentation	65
5.4.2.1	bitIndex	65
5.4.2.2	bwi	65
5.4.2.3	chipMode	65
5.4.2.4	chipType	65
5.4.2.5	cont_tim	65
5.4.2.6	conversion	65
5.4.2.7	data	65
5.4.2.8	data	65
5.4.2.9	dataIndex	65
5.4.2.10	etu	65
5.4.2.11	len	66
5.4.2.12	mode	66

5.4.2.13	parityCount	66
5.4.2.14	recExtraGuardTime	66
5.4.2.15	send_data	66
5.4.2.16	sendLen	66
5.4.2.17	sendSeqData	66
5.4.2.18	SessionATR	66
5.4.2.19	SessionATR_Size	66
5.4.2.20	t0	66
5.4.2.21	timingMode	66
5.4.2.22	tmpByteWait_time	67
5.5	p3_fw_ccid_exec Struct Reference	67
5.5.1	Detailed Description	67
5.5.2	Field Documentation	67
5.5.2.1	bh	67
5.5.2.2	bSlot	67
5.6	p3fw_data Struct Reference	67
5.6.1	Detailed Description	71
5.6.2	Field Documentation	71
5.6.2.1	aBuffer	71
5.6.2.2	bAuth	71
5.6.2.3	bCLLastNotifyEvent	71
5.6.2.4	bConfig	71
5.6.2.5	bDoPoll	71
5.6.2.6	bfl	72
5.6.2.7	bIsBusy	72
5.6.2.8	bKeyType	72
5.6.2.9	bMaxSlots	72
5.6.2.10	bMFCKey	72
5.6.2.11	bMFCKeyLength	72
5.6.2.12	cc	72
5.6.2.13	ccActiveSlotNum	72
5.6.2.14	cl	72
5.6.2.15	clActiveSlotNum	72
5.6.2.16	comm	73
5.6.2.17	config	73
5.6.2.18	dwIndex	73
5.6.2.19	dwLength	73
5.6.2.20	dwWrittenLength	73
5.6.2.21	eBalCfg	73
5.6.2.22	eClSlotsType	73
5.6.2.23	eExtIf	73
5.6.2.24	eMode	73
5.6.2.25	ePcScMode	73
5.6.2.26	eReaderIC	74
5.6.2.27	eSam	74
5.6.2.28	in	74
5.6.2.29	ipc	74
5.6.2.30	out	74
5.6.2.31	pBal	74
5.6.2.32	pcsc	74

5.6.2.33	sBalI2c	74
5.6.2.34	sBalSerial	74
5.6.2.35	sBalSpi	74
5.6.2.36	sCidManager	74
5.6.2.37	send	75
5.6.2.38	sKeyEntry	75
5.6.2.39	sKeyKUC	75
5.6.2.40	sKeyStore	75
5.6.2.41	sKeyVersion	75
5.6.2.42	slots	75
5.6.2.43	sMultexBal	75
5.6.2.44	sQueueCCIDExec	75
5.7	p3fw_hal_buffer Struct Reference	76
5.7.1	Detailed Description	76
5.7.2	Field Documentation	76
5.7.2.1	rx	76
5.7.2.2	tx	76
5.8	sam_t1_param Struct Reference	76
5.8.1	Detailed Description	77
5.8.2	Field Documentation	77
5.8.2.1	BWI_CWI	77
5.8.2.2	ClockStop	77
5.8.2.3	FI_DI	77
5.8.2.4	GuardTime	77
5.8.2.5	IFSC	77
6	File Documentation	79
6.1	include/debug_frmwrk.h File Reference	79
6.1.1	Define Documentation	82
6.1.1.1	_DBC	82
6.1.1.2	_DBD	82
6.1.1.3	_DBD16	82
6.1.1.4	_DBD32	82
6.1.1.5	_DBG	82
6.1.1.6	_DBG_	82
6.1.1.7	_DBGH_	82
6.1.1.8	_DBGHL_	82
6.1.1.9	_DBGL	82
6.1.1.10	_DBGL_	82
6.1.1.11	_DBH	83
6.1.1.12	_DBH16	83
6.1.1.13	_DBH16L	83
6.1.1.14	_DBH32	83
6.1.1.15	_DBH32L	83
6.1.1.16	_DG	83
6.1.1.17	DEBUG_UART_PORT	83
6.1.2	Function Documentation	83
6.1.2.1	debug_frmwrk_init	83
6.1.2.2	UARTGetChar	84
6.1.2.3	UARTPutChar	84

6.1.2.4	UARTPutDec	84
6.1.2.5	UARTPutDec16	84
6.1.2.6	UARTPutDec32	85
6.1.2.7	UARTPutHex	85
6.1.2.8	UARTPutHex16	85
6.1.2.9	UARTPutHex32	86
6.1.2.10	UARTPutHexMulti	86
6.1.2.11	UARTPuts	86
6.1.2.12	UARTPuts_	86
6.1.3	Variable Documentation	87
6.1.3.1	_db_char	87
6.1.3.2	_db_dec	87
6.1.3.3	_db_dec_16	87
6.1.3.4	_db_dec_32	87
6.1.3.5	_db_get_char	87
6.1.3.6	_db_hex	87
6.1.3.7	_db_hex_16	87
6.1.3.8	_db_hex_32	87
6.1.3.9	_db_mhex	87
6.1.3.10	_db_msg	88
6.1.3.11	_db_msg_	88
6.2	include/p3fw_apdu.h File Reference	88
6.3	include/p3fw_ccid.h File Reference	89
6.4	include/p3fw_dm.h File Reference	91
6.5	include/p3fw_fw.h File Reference	95
6.6	include/p3fw_hw.h File Reference	99
6.7	include/p3fw_psc.h File Reference	100
6.8	include/p3fw_pins.h File Reference	101
6.9	include/p3fw_samt1.h File Reference	101
6.9.1	Define Documentation	102
6.9.1.1	P2_FW_SAM_FRAME_APDU	102
6.9.1.2	P2_FW_SAM_FRAME_T1	102
6.9.1.3	P2_FW_SAM_MODE_PPS	102
6.9.1.4	P2_FW_SAM_MODE_X	103
6.9.1.5	P2_FW_TIMING_MODE_COM	103
6.9.1.6	P2_FW_TIMING_MODE_FDT	103
6.9.1.7	P2_FW_TIMING_MODE_NONE	103
6.9.1.8	p3fw_samt1_deinit	103
6.9.1.9	p3fw_samt1_get_cont_tim	103
6.9.1.10	p3fw_samt1_get_timing_mode	103
6.9.1.11	p3fw_samt1_getATR	103
6.9.1.12	p3fw_samt1_is_busy	103
6.9.1.13	p3fw_samt1_is_sam_inserted	103
6.9.1.14	p3fw_samt1_receive	103
6.9.1.15	p3fw_samt1_send	104
6.9.1.16	p3fw_samt1_set_etu	104
6.9.1.17	p3fw_samt1_set_timing_mode	104
6.9.1.18	p3fw_samt1_start	104
6.9.1.19	p3fw_samt1_warm_reset	104
6.9.2	Function Documentation	104

6.9.2.1	EINT3_IRQHandler	104
6.9.2.2	p2_fw_sam_t1_deinit	104
6.9.2.3	p2_fw_sam_t1_get_atr	104
6.9.2.4	p2_fw_sam_t1_get_cont_tim	104
6.9.2.5	p2_fw_sam_t1_get_timing_mode	104
6.9.2.6	p2_fw_sam_t1_init	105
6.9.2.7	p2_fw_sam_t1_is_busy	105
6.9.2.8	p2_fw_sam_t1_is_power_off	105
6.9.2.9	p2_fw_sam_t1_is_sam_inserted	105
6.9.2.10	p2_fw_sam_t1_pps	105
6.9.2.11	p2_fw_sam_t1_prepare_pps	105
6.9.2.12	p2_fw_sam_t1_receive	105
6.9.2.13	p2_fw_sam_t1_send	105
6.9.2.14	p2_fw_sam_t1_set_bwi_cwi	105
6.9.2.15	p2_fw_sam_t1_set_etu	105
6.9.2.16	p2_fw_sam_t1_set_my_debug	105
6.9.2.17	p2_fw_sam_t1_set_rec_extraGuardTime	106
6.9.2.18	p2_fw_sam_t1_set_timing_mode	106
6.9.2.19	p2_fw_sam_t1_start	106
6.9.2.20	p2_fw_sam_t1_warm_reset	106
6.9.2.21	p3fw_samt1_exchange	106
6.9.2.22	p3fw_samt1_init	106
6.9.2.23	p3fw_samt1_power_on	106
6.9.2.24	p3fw_samt1_set_param	106
6.9.2.25	TIMER0_IRQHandler	106
6.9.2.26	TIMER2_IRQHandler	106
6.10	src/debug_frmwrk.c File Reference	107
6.10.1	Detailed Description	108
6.10.2	Define Documentation	109
6.10.2.1	_DEBUG_FRMWRK_	109
6.10.3	Function Documentation	109
6.10.3.1	debug_frmwrk_init	109
6.10.3.2	UARTGetChar	109
6.10.3.3	UARTPutChar	110
6.10.3.4	UARTPutDec	110
6.10.3.5	UARTPutDec16	110
6.10.3.6	UARTPutDec32	111
6.10.3.7	UARTPutHex	111
6.10.3.8	UARTPutHex16	111
6.10.3.9	UARTPutHex32	111
6.10.3.10	UARTPutHexMulti	112
6.10.3.11	UARTPuts	112
6.10.3.12	UARTPuts_	112
6.10.4	Variable Documentation	113
6.10.4.1	_db_char	113
6.10.4.2	_db_dec	113
6.10.4.3	_db_dec_16	113
6.10.4.4	_db_dec_32	113
6.10.4.5	_db_get_char	113
6.10.4.6	_db_hex	113

6.10.4.7	_db_hex_16	113
6.10.4.8	_db_hex_32	113
6.10.4.9	_db_mhex	113
6.10.4.10	_db_msg	113
6.10.4.11	_db_msg_	114
6.11	src/p3fw.c File Reference	114
6.11.1	Define Documentation	115
6.11.1.1	P3FW_DIP_CFG_BAL_I2C	115
6.11.1.2	P3FW_DIP_CFG_BAL_SPI	115
6.11.1.3	P3FW_DIP_CFG_BAL_UART	115
6.11.1.4	P3FW_DIP_CFG_SAM_IN_X	115
6.11.1.5	P3FW_DIP_CFG_SAM_NO	115
6.11.1.6	P3FW_DIP_CFG_SAM_NON_X	115
6.11.1.7	P3FW_DIP_EXT_IF_ETHERNET	115
6.11.1.8	P3FW_DIP_EXT_IF_RS232	115
6.11.1.9	P3FW_DIP_EXT_IF_RS485	116
6.11.1.10	P3FW_DIP_EXT_IF_USB	116
6.11.1.11	P3FW_DIP_MODE_ACT_BOOTLOADER	116
6.11.1.12	P3FW_DIP_MODE_CFG_OVERWRITE	116
6.11.1.13	P3FW_DIP_MODE_DEMO	116
6.11.1.14	P3FW_DIP_MODE_PCSC	116
6.11.1.15	P3FW_TASK_START_ERROR	116
6.11.1.16	P3FW_TASK_START_EXECUTE	116
6.11.1.17	P3FW_TASK_START_POLL	116
6.11.2	Function Documentation	116
6.11.2.1	main	116
6.11.2.2	p3fw_dump_regs	117
6.12	src/p3fw_bsp.c File Reference	117
6.13	src/p3fw_ccid.c File Reference	117
6.14	src/p3fw_ccid_bh.c File Reference	119
6.15	src/p3fw_ccid_th.c File Reference	120
6.16	src/p3fw_demo.c File Reference	120
6.16.1	Define Documentation	121
6.16.1.1	P3FW_DEMO_ATQB_LEN	121
6.16.1.2	P3FW_DEMO_RX_BUFFER	121
6.16.1.3	P3FW_DEMO_TX_BUFFER	121
6.16.1.4	P3FW_DEMO_UID_LEN	122
6.17	src/p3fw_dm.c File Reference	122
6.18	src/p3fw_dm_cc.c File Reference	123
6.19	src/p3fw_dm_cid.c File Reference	124
6.20	src/p3fw_dm_hal.c File Reference	124
6.20.1	Variable Documentation	125
6.20.1.1	aHalBuffers	125
6.21	src/p3fw_dm_keystore.c File Reference	125
6.22	src/p3fw_dm_l3.c File Reference	126
6.23	src/p3fw_dm_l4.c File Reference	127
6.24	src/p3fw_dm_l4a.c File Reference	127
6.25	src/p3fw_dm_ro.c File Reference	128
6.26	src/p3fw_dm_xchg.c File Reference	129
6.27	src/p3fw_error_mode.c File Reference	130

6.28	src/p3fw_ext_if_usb.c File Reference	130
6.28.1	Define Documentation	131
6.28.1.1	P3FW_MAX_SLOT_INDEX	131
6.28.1.2	P3FW_NAME_USB_OFFSET	131
6.28.1.3	P3FW_SERNUM_USB_OFFSET	132
6.28.1.4	P3FW_USB_CONTROL_REQUEST_ABORT	132
6.28.1.5	P3FW_USB_CONTROL_REQUEST_GET_CLOCK_FREQUENCIES	132
6.28.1.6	P3FW_USB_CONTROL_REQUEST_GET_DATA_RATES	132
6.28.1.7	P3FW_USB_LE_DWORD	132
6.28.1.8	P3FW_USB_LE_WORD	132
6.28.1.9	P3FW_USB_MAX_PACKET_SIZE	132
6.29	src/p3fw_flash.c File Reference	132
6.29.1	Define Documentation	134
6.29.1.1	IAP_BLANK_CHECK_SECTOR	134
6.29.1.2	IAP_COMPARE	134
6.29.1.3	IAP_COPY_RAM_TO_FLASH	134
6.29.1.4	IAP_ERASE_SECTOR	134
6.29.1.5	IAP_LOCATION	134
6.29.1.6	IAP_PREP_SECTORS	134
6.29.1.7	IAP_READ_BOOT_CODE_VER	134
6.29.1.8	IAP_READ_DEV_SER_NUM	134
6.29.1.9	IAP_READ_ID	135
6.29.1.10	IAP_REINVOKE	135
6.29.1.11	IAP_SEC_29	135
6.29.1.12	P3FW_CFG_COUNT	135
6.29.1.13	P3FW_CFG_ENTRY_OFFSET	135
6.29.1.14	P3FW_CFG_ENTRY_SIZE	135
6.29.1.15	P3FW_CFG_MAX	135
6.29.1.16	P3FW_CFG_S1	135
6.29.1.17	P3FW_CFG_S2	135
6.29.1.18	P3FW_FLASH_BUFFER	135
6.29.2	Typedef Documentation	136
6.29.2.1	IAP	136
6.30	src/p3fw_psc.c File Reference	136
6.31	src/p3fw_psc_mem_cards.c File Reference	137
6.32	src/p3fw_poll.c File Reference	137
6.33	src/p3fw_samt1.c File Reference	138
6.33.1	Define Documentation	140
6.33.1.1	BITBAND_PERI	140
6.33.1.2	BITBAND_PERI_BASE	140
6.33.1.3	BITBAND_PERI_REF	140
6.33.1.4	P2_FW_SAM_CLOCK_LEN_NS	140
6.33.1.5	P2_FW_SAM_CONVERSION_DIRECT	141
6.33.1.6	P2_FW_SAM_CONVERSION_INVERSE	141
6.33.1.7	P2_FW_SAM_CONVERSION_UNKNOWN	141
6.33.1.8	P2_FW_SAM_DEBUG	141
6.33.1.9	P2_FW_SAM_DEFAULT_BWI	141
6.33.1.10	P2_FW_SAM_MAX_ATR_SIZE	141

6.33.1.11	P2_FW_SAM_MAX_REC_DATALEN	141
6.33.1.12	P2_FW_SAM_MODE_IDLE	141
6.33.1.13	P2_FW_SAM_MODE_POWER_OFF	141
6.33.1.14	P2_FW_SAM_MODE_RECEIVE	141
6.33.1.15	P2_FW_SAM_MODE_RESET	141
6.33.1.16	P2_FW_SAM_MODE_SEND	142
6.33.1.17	P2_FW_SAM_MODE_WAIT_START_BIT	142
6.33.1.18	P2_FW_SAM_PWM_CH0_MATCH_VALUE	142
6.33.1.19	P2_FW_SAM_PWM_CH1_MATCH_VALUE	142
6.33.1.20	P2_FW_SAM_T1_CLOCK_CONFIG	142
6.33.1.21	P2_FW_SAM_TIMER_PRESCALE_VALUE	142
6.33.1.22	T1_BYTE_WAIT_TIME	142
6.33.1.23	TIMER0_IR	142
6.33.1.24	TIMER0_IR_MR0	142
6.33.1.25	TIMER0_TCR	142
6.33.1.26	TIMER0_TCR_ENABLE	143
6.33.1.27	TIMER0_TCR_RESET	143
6.33.2	Function Documentation	143
6.33.2.1	EINT3_IRQHandler	143
6.33.2.2	p2_fw_sam_t1_deinit	143
6.33.2.3	p2_fw_sam_t1_get_atr	143
6.33.2.4	p2_fw_sam_t1_get_cont_tim	143
6.33.2.5	p2_fw_sam_t1_get_timing_mode	143
6.33.2.6	p2_fw_sam_t1_init	143
6.33.2.7	p2_fw_sam_t1_is_busy	143
6.33.2.8	p2_fw_sam_t1_is_power_off	143
6.33.2.9	p2_fw_sam_t1_is_sam_inserted	144
6.33.2.10	p2_fw_sam_t1_pps	144
6.33.2.11	p2_fw_sam_t1_prepare_pps	144
6.33.2.12	p2_fw_sam_t1_receive	144
6.33.2.13	p2_fw_sam_t1_send	144
6.33.2.14	p2_fw_sam_t1_set_bwi_cwi	144
6.33.2.15	p2_fw_sam_t1_set_etu	144
6.33.2.16	p2_fw_sam_t1_set_my_debug	144
6.33.2.17	p2_fw_sam_t1_set_rec_extraGuardTime	144
6.33.2.18	p2_fw_sam_t1_set_timing_mode	144
6.33.2.19	p2_fw_sam_t1_start	144
6.33.2.20	p2_fw_sam_t1_warm_reset	145
6.33.2.21	TIMER0_IRQHandler	145
6.33.2.22	TIMER2_IRQHandler	145
6.34	src/p3fw_samt1_wrap.c File Reference	145
6.34.1	Function Documentation	146
6.34.1.1	p3fw_samt1_exchange	146
6.34.1.2	p3fw_samt1_init	146
6.34.1.3	p3fw_samt1_power_on	146
6.34.1.4	p3fw_samt1_set_param	146
6.35	src/p3fw_slots.c File Reference	146
6.35.1	Define Documentation	147
6.35.1.1	P3FW_TEMP_BUFF_LEN	147
6.35.2	Variable Documentation	147

6.35.2.1 aHalBuffers	147
6.36 src/p3fw_timing.c File Reference	148

This reference manual documents the SW architecture of the Pegoda RD710/RD852 reader firmware.

DISCLAIMER OF WARRANTIES:

YOU AGREE THAT NXP HAS MADE NO EXPRESS WARRANTIES TO YOU REGARDING THE SOFTWARE AND THAT THE SOFTWARE IS BEING PROVIDED TO YOU "AS IS" WITHOUT WARRANTY OF ANY KIND. NXP DISCLAIMS ALL WARRANTIES WITH REGARD TO THE SOFTWARE, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, MERCHANTABILITY QUALITY OR NONINFRINGEMENT OF THIRD PARTY RIGHTS.

Some states or jurisdictions do not allow the exclusion of implied warranties so the above limitations or parts of it may not apply to you.

LIMITATION OF LIABILITY:

IN NO EVENT WILL NXP BE LIABLE TO YOU FOR ANY LOSS OF USE, INTERRUPTION OF BUSINESS, OR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND (INCLUDING LOST PROFITS) REGARDLESS OF THE FORM OF ACTION WHETHER IN CONTRACT, TORT (INCLUDING NEGLIGENCE), STRICT PRODUCT LIABILITY OR OTHERWISE, EVEN IF ENDUSER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Some states or jurisdictions do not allow the exclusion or limitation of incidental or Consequential damages, so the above limitation, exclusion or parts of it may not apply to you.

Revision History

VERSION	DATE	DESCRIPTION
2.0	12.06.2012	General update for new Firmware
1.0	10.11.2010	First Release

Chapter 1

Module Index

1.1 Modules

Here is a list of all modules:

APDUs Defines	7
CCID functions	9
Mode Functions	19
Firmware	37
Initialization	53
PCSCs	55

Chapter 2

Data Structure Index

2.1 Data Structures

Here are the data structures with brief descriptions:

_p3fw_cc_slot	59
_p3fw_cl_slot (Stores state of a contact less slot)	60
cpot_atr_frame	63
p2_fw_SAM_ctrl	64
p3_fw_ccid_exec (Job parameters for)	67
p3fw_data (Stores all reader global settings and variables)	67
p3fw_hal_buffer (Every HAL requires buffer for sending and receiving)	76
sam_t1_param	76

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

include/debug_frmwrk.h	79
include/p3fw_apdu.h	88
include/p3fw_ccid.h	89
include/p3fw_dm.h	91
include/p3fw_fw.h	95
include/p3fw_hw.h	99
include/p3fw_pcsc.h	100
include/p3fw_pins.h	101
include/p3fw_samt1.h	101
src/debug_frmwrk.c (: Contains some utilities that used for debugging through UART)	107
src/p3fw.c	114
src/p3fw_bsp.c	117
src/p3fw_ccid.c	117
src/p3fw_ccid_bh.c	119
src/p3fw_ccid_th.c	120
src/p3fw_demo.c	120
src/p3fw_dm.c	122
src/p3fw_dm_cc.c	123
src/p3fw_dm_cid.c	124
src/p3fw_dm_hal.c	124
src/p3fw_dm_keystore.c	125
src/p3fw_dm_l3.c	126
src/p3fw_dm_l4.c	127
src/p3fw_dm_l4a.c	127
src/p3fw_dm_ro.c	128
src/p3fw_dm_xchg.c	129
src/p3fw_error_mode.c	130
src/p3fw_ext_if_usb.c	130

src/p3fw_flash.c	132
src/p3fw_pcsc.c	136
src/p3fw_pcsc_mem_cards.c	137
src/p3fw_poll.c	137
src/p3fw_samt1.c	138
src/p3fw_samt1_wrap.c	145
src/p3fw_slots.c	146
src/p3fw_timing.c	148

Chapter 4

Module Documentation

4.1 APDUs Defines

Defines

- `#define P3FW_APDU_PAYLOAD 0`
- `#define P3FW_APDU_CC_EXT_MANAGE_SESSION 0x00`
- `#define P3FW_APDU_CC_EXT_TRANS_EXCHANGE 0x01`
- `#define P3FW_APDU_CC_EXT_SWITCH_PROTOCOL 0x02`

ISO7816 APDUs offsets

- `#define P3FW_APDU_CLASS 0`
- `#define P3FW_APDU_INS 1`
- `#define P3FW_APDU_P1 2`
- `#define P3FW_APDU_P2 3`
- `#define P3FW_APDU_Lc 4`
- `#define P3FW_APDU_Le 4`

PCSC Memory Cards Extensions

- `#define P3FW_APDU_CC_EXT_INS 0xC2`
- `#define P3FW_APDU_GET_DATA_INS 0xCA`
- `#define P3FW_APDU_LOAD_KEY_INS 0x82`
- `#define P3FW_APDU_G_AUTH_CMD_INS 0x86`
- `#define P3FW_APDU_READ_BIN 0xB0`
- `#define P3FW_APDU_UPDATE_BIN 0xD6`

4.1.1 Define Documentation

4.1.1.1 **#define P3FW_APDU_CC_EXT_INS 0xC2**

Definition at line 51 of file p3fw_apdu.h.

4.1.1.2 **#define P3FW_APDU_CC_EXT_MANAGE_SESSION 0x00**

Definition at line 43 of file p3fw_apdu.h.

4.1.1.3 **#define P3FW_APDU_CC_EXT_SWITCH_PROTOCOL 0x02**

Definition at line 45 of file p3fw_apdu.h.

4.1.1.4 **#define P3FW_APDU_CC_EXT_TRANS_EXCHANGE 0x01**

Definition at line 44 of file p3fw_apdu.h.

4.1.1.5 **#define P3FW_APDU_CLASS 0**

APDU Class Offset

Definition at line 32 of file p3fw_apdu.h.

4.1.1.6 **#define P3FW_APDU_G_AUTH_CMD_INS 0x86**

PCSC Extension: Authenticate Command

Definition at line 54 of file p3fw_apdu.h.

4.1.1.7 **#define P3FW_APDU_GET_DATA_INS 0xCA**

PCSC Extension: GetData

Definition at line 52 of file p3fw_apdu.h.

4.1.1.8 **#define P3FW_APDU_INS 1**

APDU Instruction Offset

Definition at line 33 of file p3fw_apdu.h.

4.1.1.9 **#define P3FW_APDU_Lc 4**

APDU Lc Offset

Definition at line 36 of file p3fw_apdu.h.

4.1.1.10 #define P3FW_APDU_Lc 4

APDU Lc Offset

Definition at line 37 of file p3fw_apdu.h.

4.1.1.11 #define P3FW_APDU_LOAD_KEY_INS 0x82

PCSC Extension: LoadKey

Definition at line 53 of file p3fw_apdu.h.

4.1.1.12 #define P3FW_APDU_P1 2

APDU P1 Offset

Definition at line 34 of file p3fw_apdu.h.

4.1.1.13 #define P3FW_APDU_P2 3

APDU P2 Offset

Definition at line 35 of file p3fw_apdu.h.

4.1.1.14 #define P3FW_APDU_PAYLOAD 0

APDU payload Offset

Definition at line 40 of file p3fw_apdu.h.

4.1.1.15 #define P3FW_APDU_READ_BIN 0xB0

PCSC Extension: Read Binary

Definition at line 55 of file p3fw_apdu.h.

4.1.1.16 #define P3FW_APDU_UPDATE_BIN 0xD6

PCSC Extension: Update Binary

Definition at line 56 of file p3fw_apdu.h.

4.2 CCID functions

Defines

- [#define P3FW_CCID_BULK_HEADER 0x0A](#)

- #define [P3FW_CCID_HEADER_MESSAGE_TYPE](#) 0x00
- #define [P3FW_CCID_HEADER_LENGTH_BYTE_1](#) 0x01
- #define [P3FW_CCID_HEADER_LENGTH_BYTE_2](#) 0x02
- #define [P3FW_CCID_HEADER_LENGTH_BYTE_3](#) 0x03
- #define [P3FW_CCID_HEADER_LENGTH_BYTE_4](#) 0x04
- #define [P3FW_CCID_HEADER_SLOT](#) 0x05
- #define [P3FW_CCID_HEADER_SEQ](#) 0x06
- #define [P3FW_CCID_HEADER_MSG_BYTE_1](#) 0x07
- #define [P3FW_CCID_HEADER_MSG_BYTE_2](#) 0x08
- #define [P3FW_CCID_HEADER_MSG_BYTE_3](#) 0x09
- #define [P3FW_CCID_RDR_TO_PC_SLOT_DATA_BLOCK](#) 0x80
- #define [P3FW_CCID_RDR_TO_PC_SLOT_STATUS](#) 0x81
- #define [P3FW_CCID_RDR_TO_PC_PARAMETERS](#) 0x82
- #define [P3FW_CCID_RDR_TO_PC_ESCAPE](#) 0x83
- #define [P3FW_CCID_STATUS_CMD_FAILED](#) 0x40
- #define [P3FW_CCID_ERROR_SLOT_BUSY](#) 0xE0
- #define [P3FW_CCID_ERROR_SLOT_NOT_EXIST](#) 0x05
- #define [P3FW_CCID_ERROR_SLOT_ICC_MUTE](#) 0xFE
- #define [P3FW_CCID_ERROR_SLOT_XFR_OVERRUN](#) 0xFC
- #define [P3FW_CCID_ERROR_SLOT_CMD_NOT_SUPPORTED](#) 0x00
- #define [P3FW_CCID_ERROR_SLOT_HW_ERROR](#) 0xFB
- #define [P3FW_CCID_BULK_OUT_REQ_ICCPowerON](#) 0x62
- #define [P3FW_CCID_BULK_OUT_REQ_ICCPowerOFF](#) 0x63
- #define [P3FW_CCID_BULK_OUT_REQ_GETSLOTSTATUS](#) 0x65
- #define [P3FW_CCID_BULK_OUT_REQ_XFRBLOCK](#) 0x6F
- #define [P3FW_CCID_BULK_OUT_REQ_GETPARAMETERS](#) 0x6C
- #define [P3FW_CCID_BULK_OUT_REQ_SETPARAMETERS](#) 0x61
- #define [P3FW_CCID_BULK_OUT_REQ_ESCAPE](#) 0x6B
- #define [P3FW_CCID_INT_IN_NOTIFY_SLOT_CHANGE](#) 0x50
- #define [P3FW_CCID_CARD_IN_SLOT](#) 0x01
- #define [P3FW_CCID_CARD_OUT_SLOT](#) 0x00

Functions

- void [p3fw_ccid_th_dispatch](#) (void)
Main function for top half CCID handler.
- void [p3fw_ccid_send_frame](#) (uint8_t bMessageType, uint8_t bByte1, uint8_t bByte2, uint8_t bByte3)
Sends CCID frame over USB to the driver.
- uint8_t * [p3fw_ccid_get_output_payload_buffer](#) ()
Returns a pointer to CCID internal output buffer.
- void [p3fw_ccid_set_output_payload_length](#) (uint32_t dwPayloadLength)
Sets the payload length of the output CCID message.

- `uint8_t * p3fw_ccid_get_input_payload_buffer ()`
Returns a pointer to CCID internal input buffer.
- `uint32_t p3fw_ccid_get_input_payload_length ()`
Gets the payload length of the input CCID message.
- `uint8_t p3fw_ccid_input_header_get_byte (uint8_t bByte)`
Gets one byte from input header.
- `void p3fw_ccid_bh_get_slot_status (uint8_t bCCIDSlotIndex)`
Returns /c bCCIDSlotIndex status.
- `void p3fw_ccid_bh_icc_power_on (uint8_t bCCIDSlotIndex)`
Card Power On.
- `void p3fw_ccid_bh_icc_power_off (uint8_t bCCIDSlotIndex)`
Card Power Off.
- `void p3fw_ccid_bh_get_parameters (uint8_t bCCIDSlotIndex)`
CCID Get Parameters.
- `void p3fw_ccid_bh_set_parameters (uint8_t bCCIDSlotIndex)`
CCID Set Parameters.
- `void p3fw_ccid_bh_escape (uint8_t bCCIDSlotIndex)`
CCID Escape Interface (SCardControl)
- `void p3fw_ccid_bh_xfer (uint8_t bCCIDSlotIndex)`
CCID Transfer command.
- `void p3fw_ccid_send_notify ()`
Sends slot status notification through INT end-point.

4.2.1 Detailed Description

Supporting function.

4.2.2 Define Documentation

4.2.2.1 `#define P3FW_CCID_BULK_HEADER 0x0A`

CCID Header Size

Definition at line 27 of file `p3fw_ccid.h`.

4.2.2.2 #define P3FW_CCID_BULK_OUT_REQ_ESCAPE 0x6B

ICC Power On Command

Definition at line 60 of file p3fw_ccid.h.

4.2.2.3 #define P3FW_CCID_BULK_OUT_REQ_GETPARAMETERS 0x6C

Get Parameters Command

Definition at line 58 of file p3fw_ccid.h.

4.2.2.4 #define P3FW_CCID_BULK_OUT_REQ_GETSLOTSTATUS 0x65

Get Slot Status Command

Definition at line 56 of file p3fw_ccid.h.

4.2.2.5 #define P3FW_CCID_BULK_OUT_REQ_ICCPowerOFF 0x63

ICC Power Off Command

Definition at line 55 of file p3fw_ccid.h.

4.2.2.6 #define P3FW_CCID_BULK_OUT_REQ_ICCPowerON 0x62

ICC Power On Command

Definition at line 54 of file p3fw_ccid.h.

4.2.2.7 #define P3FW_CCID_BULK_OUT_REQ_SETPARAMETERS 0x61

ICC Power On Command

Definition at line 59 of file p3fw_ccid.h.

4.2.2.8 #define P3FW_CCID_BULK_OUT_REQ_XFRBLOCK 0x6F

Transfer Block Command

Definition at line 57 of file p3fw_ccid.h.

4.2.2.9 #define P3FW_CCID_CARD_IN_SLOT 0x01

Slot is occupied

Definition at line 64 of file p3fw_ccid.h.

4.2.2.10 #define P3FW_CCID_CARD_OUT_SLOT 0x00

Slot is not occupied

Definition at line 65 of file p3fw_ccid.h.

4.2.2.11 #define P3FW_CCID_ERROR_SLOT_BUSY 0xE0

Error Code: Slot is busy

Definition at line 47 of file p3fw_ccid.h.

4.2.2.12 #define P3FW_CCID_ERROR_SLOT_CMD_NOT_SUPPORTED 0x00

Error Code: Command not supported

Definition at line 51 of file p3fw_ccid.h.

4.2.2.13 #define P3FW_CCID_ERROR_SLOT_HW_ERROR 0xFB

Error Code: Hardware error

Definition at line 52 of file p3fw_ccid.h.

4.2.2.14 #define P3FW_CCID_ERROR_SLOT_ICC_MUTE 0xFE

Error Code: ICC is mute

Definition at line 49 of file p3fw_ccid.h.

4.2.2.15 #define P3FW_CCID_ERROR_SLOT_NOT_EXIST 0x05

Error Code: Slot does not exist

Definition at line 48 of file p3fw_ccid.h.

4.2.2.16 #define P3FW_CCID_ERROR_SLOT_XFR_OVERRUN 0xFC

Error Code: Buffer overrun

Definition at line 50 of file p3fw_ccid.h.

4.2.2.17 #define P3FW_CCID_HEADER_LENGTH_BYTE_1 0x01

CCID Header - Byte 2 - Length 1 (LSB)

Definition at line 30 of file p3fw_ccid.h.

4.2.2.18 #define P3FW_CCID_HEADER_LENGTH_BYTE_2 0x02

CCID Header - Byte 3 - Length 2

Definition at line 31 of file p3fw_ccid.h.

4.2.2.19 #define P3FW_CCID_HEADER_LENGTH_BYTE_3 0x03

CCID Header - Byte 4 - Length 3

Definition at line 32 of file p3fw_ccid.h.

4.2.2.20 #define P3FW_CCID_HEADER_LENGTH_BYTE_4 0x04

CCID Header - Byte 5 - Length 4 (MSB)

Definition at line 33 of file p3fw_ccid.h.

4.2.2.21 #define P3FW_CCID_HEADER_MESSAGE_TYPE 0x00

CCID Header - Byte 1 - Message Type

Definition at line 29 of file p3fw_ccid.h.

4.2.2.22 #define P3FW_CCID_HEADER_MSG_BYTE_1 0x07

CCID Header - Byte 8 - Message Byte 1

Definition at line 36 of file p3fw_ccid.h.

4.2.2.23 #define P3FW_CCID_HEADER_MSG_BYTE_2 0x08

CCID Header - Byte 9 - Message Byte 2

Definition at line 37 of file p3fw_ccid.h.

4.2.2.24 #define P3FW_CCID_HEADER_MSG_BYTE_3 0x09

CCID Header - Byte 10 - Message Byte 3

Definition at line 38 of file p3fw_ccid.h.

4.2.2.25 #define P3FW_CCID_HEADER_SEQ 0x06

CCID Header - Byte 7 - Sequence

Definition at line 35 of file p3fw_ccid.h.

4.2.2.26 #define P3FW_CCID_HEADER_SLOT 0x05

CCID Header - Byte 6 - Slot Number

Definition at line 34 of file p3fw_ccid.h.

4.2.2.27 #define P3FW_CCID_INT_IN_NOTIFY_SLOT_CHANGE 0x50

Slot Change Notify Command

Definition at line 62 of file p3fw_ccid.h.

4.2.2.28 #define P3FW_CCID_RDR_TO_PC_ESCAPE 0x83

Escape Message Replay Block

Definition at line 43 of file p3fw_ccid.h.

4.2.2.29 #define P3FW_CCID_RDR_TO_PC_PARAMETERS 0x82

Slot Parameters Replay

Definition at line 42 of file p3fw_ccid.h.

4.2.2.30 #define P3FW_CCID_RDR_TO_PC_SLOT_DATA_BLOCK 0x80

Data Message Replay Block

Definition at line 40 of file p3fw_ccid.h.

4.2.2.31 #define P3FW_CCID_RDR_TO_PC_SLOT_STATUS 0x81

Slot Status Replay

Definition at line 41 of file p3fw_ccid.h.

4.2.2.32 #define P3FW_CCID_STATUS_CMD_FAILED 0x40

Command Faild

Definition at line 45 of file p3fw_ccid.h.

4.2.3 Function Documentation**4.2.3.1 void p3fw_ccid_bh_escape (uint8_t bCCIDSlotIndex)**

CCID Escape Interface (SCardControl)

Parameters

in	<i>bCCID-SlotIndex</i>	slot index
----	------------------------	------------

Definition at line 288 of file p3fw_ccid_bh.c.

4.2.3.2 void p3fw_ccid_bh_get_parameters (uint8_t *bCCIDSlotIndex*)

CCID Get Parameters.

Parameters

in	<i>bCCID-SlotIndex</i>	slot index
----	------------------------	------------

Definition at line 155 of file p3fw_ccid_bh.c.

4.2.3.3 void p3fw_ccid_bh_get_slot_status (uint8_t *bCCIDSlotIndex*)

Returns /c *bCCIDSlotIndex* status.

Parameters

in	<i>bCCID-SlotIndex</i>	slot index
----	------------------------	------------

Definition at line 27 of file p3fw_ccid_bh.c.

4.2.3.4 void p3fw_ccid_bh_icc_power_off (uint8_t *bCCIDSlotIndex*)

Card Power Off.

Parameters

in	<i>bCCID-SlotIndex</i>	slot index
----	------------------------	------------

Definition at line 98 of file p3fw_ccid_bh.c.

4.2.3.5 void p3fw_ccid_bh_icc_power_on (uint8_t *bCCIDSlotIndex*)

Card Power On.

Parameters

in	<i>bCCID-SlotIndex</i>	slot index
----	------------------------	------------

Definition at line 70 of file p3fw_ccid_bh.c.

4.2.3.6 void p3fw_ccid_bh_set_parameters (uint8_t *bCCIDSlotIndex*)

CCID Set Parameters.

Parameters

in	<i>bCCID-SlotIndex</i>	slot index
----	------------------------	------------

Definition at line 216 of file p3fw_ccid_bh.c.

4.2.3.7 void p3fw_ccid_bh_xfer (uint8_t *bCCIDSlotIndex*)

CCID Transfer command.

Parameters

in	<i>bCCID-SlotIndex</i>	slot index
----	------------------------	------------

Definition at line 293 of file p3fw_ccid_bh.c.

4.2.3.8 uint8_t* p3fw_ccid_get_input_payload_buffer ()

Returns a pointer to CCID internal input buffer.

Returns

Pointer to CCID internal output buffer

Definition at line 236 of file p3fw_ccid.c.

4.2.3.9 uint32_t p3fw_ccid_get_input_payload_length ()

Gets the payload length of the input CCID message.

Definition at line 241 of file p3fw_ccid.c.

4.2.3.10 uint8_t* p3fw_ccid_get_output_payload_buffer ()

Returns a pointer to CCID internal output buffer.

Returns

Pointer to CCID internal output buffer

Definition at line 226 of file p3fw_ccid.c.

4.2.3.11 uint8_t p3fw_ccid_input_header_get_byte (uint8_t *bByte*)

Gets one byte from input header.

Parameters

in	<i>bByte</i>	Byte to get
----	--------------	-------------

Definition at line 216 of file p3fw_ccid.c.

4.2.3.12 void p3fw_ccid_send_frame (uint8_t *bMessageType*, uint8_t *bByte1*, uint8_t *bByte2*, uint8_t *bByte3*)

Sends CCID frame over USB to the driver.

Parameters

in	<i>bMessageType</i>	USB CCID Message Type
in	<i>bByte1</i>	Message specific byte 1
in	<i>bByte2</i>	Message specific byte 2
in	<i>bByte3</i>	Message specific byte 3

Definition at line 175 of file p3fw_ccid.c.

4.2.3.13 void p3fw_ccid_send_notify ()

Sends slot status notification through INT end-point.

Definition at line 53 of file p3fw_ccid.c.

4.2.3.14 void p3fw_ccid_set_output_payload_length (uint32_t *dwPayloadLength*)

Sets the payload length of the output CCID message.

Parameters

in	<i>dwPayloadLength</i>	payload length
----	------------------------	----------------

Definition at line 231 of file p3fw_ccid.c.

4.2.3.15 void p3fw_ccid_th_dispatch (void)

Main function for top half CCID handler.

Definition at line 22 of file p3fw_ccid_th.c.

4.3 Mode Functions

Defines

- #define P3FW_DM_CID 0x90
- #define P3FW_DM_RO 0xA0
- #define P3FW_DM_HAL 0xB0
- #define P3FW_DM_L3 0xC0
- #define P3FW_DM_L4A 0xD0
- #define P3FW_DM_L4 0xE0
- #define P3FW_DM_XCHG 0xF0
- #define P3FW_DM_KSTOR 0x70
- #define P3FW_DM_CONTACT_CARD 0x80
- #define P3FW_DM_RO_LEDS_OFF 0x01
- #define P3FW_DM_RO_LEDS_ON 0x02
- #define P3FW_DM_RO_RESET 0x03
- #define P3FW_DM_RO_CONF_OVER 0x04
- #define P3FW_DM_RO_SET_CONF 0x05
- #define P3FW_DM_RO_GET_CONF 0x0C
- #define P3FW_DM_RO_GET_STATUS 0x06
- #define P3FW_DM_RO_READ_REG 0x07
- #define P3FW_DM_RO_WRITE_REG 0x08
- #define P3FW_DM_RO_FIELD_ON 0x09
- #define P3FW_DM_RO_FIELD_OFF 0x0A
- #define P3FW_DM_RO_FIELD_RESET 0x0B
- #define P3FW_DM_RO_SET_PCSC_MODE 0x0D
- #define P3FW_DM_RO_TEST_MODE 0x0E
- #define P3FW_DM_HAL_INIT 0x01
- #define P3FW_DM_HAL_XCHG 0x02
- #define P3FW_DM_HAL_SET_CFG 0x03
- #define P3FW_DM_HAL_GET_CFG 0x04
- #define P3FW_DM_HAL_APP_PROT_SET 0x05
- #define P3FW_DM_HAL_WAIT 0x06
- #define P3FW_DM_HAL_MFC_AUTH 0x07
- #define P3FW_DM_HAL_EXEC_CMD 0x08
- #define P3FW_DM_HAL_MFC_AUTH_KEY 0x09
- #define P3FW_DM_L3_INIT 0x01
- #define P3FW_DM_L3_REQA 0x02
- #define P3FW_DM_L3_WKUA 0x03
- #define P3FW_DM_L3_HLTA 0x04
- #define P3FW_DM_L3_ANTICOL 0x05
- #define P3FW_DM_L3_SELECT 0x06
- #define P3FW_DM_L3_ACT_CARD 0x07
- #define P3FW_DM_L3_XCHG 0x08
- #define P3FW_DM_L3_GET_SER 0x09
- #define P3FW_DM_L4A_INIT 0x01

- #define P3FW_DM_L4A_RATS 0x02
- #define P3FW_DM_L4A_PPS 0x03
- #define P3FW_DM_L4A_ACT_CARD 0x04
- #define P3FW_DM_L4A_GET_PROTO_PARM 0x05
- #define P3FW_DM_L4_INIT 0x01
- #define P3FW_DM_L4_SET_PROTO 0x02
- #define P3FW_DM_L4_RESET_PROTO 0x03
- #define P3FW_DM_L4_DESELECT 0x04
- #define P3FW_DM_L4_PRES_CHECK 0x05
- #define P3FW_DM_L4_XCHG 0x06
- #define P3FW_DM_L4_SET_CFG 0x07
- #define P3FW_DM_L4_GET_CFG 0x08
- #define P3FW_DM_XCHG_L3 0x01
- #define P3FW_DM_XCHG_L4 0x02
- #define P3FW_DM_XCHG_PC 0x03
- #define P3FW_DM_XCHG_RAW 0x04
- #define P3FW_DM_XCHG_MFC_AUTH 0x05
- #define P3FW_DM_XCHG_MFC_AUTH_KEY 0x06
- #define P3FW_DM_XCHG_INIT 0x07
- #define P3FW_DM_CID_GET_FREE 0x01
- #define P3FW_DM_CID_FREE 0x02
- #define P3FW_DM_CID_INIT 0x03
- #define P3FW_DM_KSTOR_INIT 0x01
- #define P3FW_DM_KSTOR_FORMAT_KEY 0x02
- #define P3FW_DM_KSTOR_SET_KEY 0x03
- #define P3FW_DM_KSTOR_SET_KEY_POS 0x04
- #define P3FW_DM_KSTOR_SET_KUC 0x05
- #define P3FW_DM_KSTOR_SET_FULL_KEY 0x07
- #define P3FW_DM_KSTOR_GET_KEY_ENTRY 0x08
- #define P3FW_DM_KSTOR_GET_KEY 0x09
- #define P3FW_DM_KSTOR_SET_CONFIG 0x0A
- #define P3FW_DM_KSTOR_GET_CONFIG 0x0B
- #define P3FW_DM_KSTOR_CHG_KUC 0x0C
- #define P3FW_DM_KSTOR_GET_KUC 0x0D
- #define P3FW_DM_KSTOR_SET_CFG_STR 0x0E
- #define P3FW_DM_KSTOR_GET_CFG_STR 0x0F
- #define P3FW_DM_CONTACTCARD_ACTIVATE_CARD 0x01
- #define P3FW_DM_CONTACTCARD_COLD_RESET 0x02
- #define P3FW_DM_CONTACTCARD_WARM_RESET 0x03
- #define P3FW_DM_CONTACTCARD_CLOCK_STOP 0x04
- #define P3FW_DM_CONTACTCARD_CLOCK_START 0x05
- #define P3FW_DM_CONTACTCARD_DEACTIVATE_CARD 0x06
- #define P3FW_DM_CONTACTCARD_PRESENCE_CHECK 0x07
- #define P3FW_DM_CONTACTCARD_TRANSMIT_DATA 0x08
- #define P3FW_DM_CONTACTCARD_PPS 0x09
- #define P3FW_DM_CHECK_STATUS(status) if (PH_ERR_SUCCESS == status) { p3fw_dm_send_frame (status, 0); return; }

- #define `P3FW_DM_CHECK_LENGTH_EQUAL`(len) if ((len != p3fw_dm_get_input_payload_length()) || (p3fw_dm_get_input_payload_length() != (p3fw_ccid_get_input_payload_length() - 4))) { p3fw_dm_send_frame (PH_COMP_BAL | PH_ERR_LENGTH_ERROR, 0); return; }
- #define `P3FW_DM_CHECK_LENGTH_LARGER`(len) if ((len > p3fw_dm_get_input_payload_length()) || (p3fw_dm_get_input_payload_length() != (p3fw_ccid_get_input_payload_length() - 4))) { p3fw_dm_send_frame (PH_COMP_BAL | PH_ERR_LENGTH_ERROR, 0); return; }

Functions

- void `p3fw_dm_xfer` ()
Direct Mode Transfer Mode.
- void `p3fw_dm_ro` ()
Direct Mode Reader Operations (also available through SCardControl/CCID Escape Interface)
- void `p3fw_dm_hal` ()
Direct Mode Hardware Abstraction Layer Functions.
- void `p3fw_dm_l3` ()
Direct Mode ISO14443-3 Functions.
- void `p3fw_dm_l4a` ()
Direct Mode ISO14443-4A Functions.
- void `p3fw_dm_l4` ()
Direct Mode ISO14443-4 Functions.
- void `p3fw_dm_xchg` ()
Direct Mode Exchange Functions.
- void `p3fw_dm_cid` ()
Direct Mode CID Management Functions.
- void `p3fw_dm_keystore` ()
Direct Mode Keystore Management Functions.
- void `p3fw_dm_contact_card` ()
Direct Mode Contact Card Functions.
- uint8_t `p3fw_dm_get_class` ()
Extracts class from the frame.
- uint8_t `p3fw_dm_get_instruction` ()

Extracts instruction from the frame.

- `uint8_t p3fw_dm_get_slot_index ()`

Extracts slot index from the frame.

- `uint16_t p3fw_dm_get_input_payload_length ()`

Gets input length.

- `uint8_t * p3fw_dm_get_input_payload ()`

Gets input payload buffer.

- `uint8_t * p3fw_dm_get_output_payload ()`

Gets output payload buffer.

- `uint8_t * p3fw_dm_get_output_pointer_uint8_t (uint16_t wDmByteIndex)`
- `uint8_t p3fw_dm_get_input_uint8_t (uint16_t wDmByteIndex)`
- `uint16_t p3fw_dm_get_input_uint16_t (uint16_t wDmByteIndex)`
- `uint32_t p3fw_dm_get_input_uint32_t (uint16_t wDmByteIndex)`
- `uint8_t * p3fw_dm_get_input_pointer_uint8_t (uint16_t wDmByteIndex)`
- `void p3fw_dm_send_frame (phStatus_t status, uint16_t wFrameLength)`

Sends frame.

4.3.1 Define Documentation

4.3.1.1 `#define P3FW_DM_CHECK_LENGTH_EQUAL(len) if ((len != p3fw_dm_get_input_payload_length()) || (p3fw_dm_get_input_payload_length() != (p3fw_ccid_get_input_payload_length() - 4))) { p3fw_dm_send_frame (PH_COMP_BAL | PH_ERR_LENGTH_ERROR, 0); return; }`

Definition at line 127 of file `p3fw_dm.h`.

4.3.1.2 `#define P3FW_DM_CHECK_LENGTH_LARGER(len) if ((len > p3fw_dm_get_input_payload_length()) || (p3fw_dm_get_input_payload_length() != (p3fw_ccid_get_input_payload_length() - 4))) { p3fw_dm_send_frame (PH_COMP_BAL | PH_ERR_LENGTH_ERROR, 0); return; }`

Definition at line 128 of file `p3fw_dm.h`.

4.3.1.3 `#define P3FW_DM_CHECK_STATUS(status) if (PH_ERR_SUCCESS == status) { p3fw_dm_send_frame (status, 0); return; }`

Definition at line 126 of file `p3fw_dm.h`.

4.3.1.4 #define P3FW_DM_CID 0x90

Direct Mode Class: Channel ID

Definition at line 29 of file p3fw_dm.h.

4.3.1.5 #define P3FW_DM_CID_FREE 0x02

Direct Mode Command: CID Free

Definition at line 98 of file p3fw_dm.h.

4.3.1.6 #define P3FW_DM_CID_GET_FREE 0x01

Direct Mode Command: CID Get Free

Definition at line 97 of file p3fw_dm.h.

4.3.1.7 #define P3FW_DM_CID_INIT 0x03

Direct Mode Command: CID Init

Definition at line 99 of file p3fw_dm.h.

4.3.1.8 #define P3FW_DM_CONTACT_CARD 0x80

Direct Mode Class: Contact Card

Definition at line 37 of file p3fw_dm.h.

4.3.1.9 #define P3FW_DM_CONTACTCARD_ACTIVATE_CARD 0x01

Direct Mode Command: ContactCard Activate Card

Definition at line 116 of file p3fw_dm.h.

4.3.1.10 #define P3FW_DM_CONTACTCARD_CLOCK_START 0x05

Direct Mode Command: ContactCard Clock Start

Definition at line 120 of file p3fw_dm.h.

4.3.1.11 #define P3FW_DM_CONTACTCARD_CLOCK_STOP 0x04

Direct Mode Command: ContactCard Clock Stop

Definition at line 119 of file p3fw_dm.h.

4.3.1.12 #define P3FW_DM_CONTACTCARD_COLD_RESET 0x02

Direct Mode Command: ContactCard Cold Reset

Definition at line 117 of file p3fw_dm.h.

4.3.1.13 #define P3FW_DM_CONTACTCARD_DEACTIVATE_CARD 0x06

Direct Mode Command: ContactCard Deactivate Card

Definition at line 121 of file p3fw_dm.h.

4.3.1.14 #define P3FW_DM_CONTACTCARD_PPS 0x09

Direct Mode Command: ContactCard Send PPS

Definition at line 124 of file p3fw_dm.h.

4.3.1.15 #define P3FW_DM_CONTACTCARD_PRESENCE_CHECK 0x07

Direct Mode Command: ContactCard Presence Check

Definition at line 122 of file p3fw_dm.h.

4.3.1.16 #define P3FW_DM_CONTACTCARD_TRANSMIT_DATA 0x08

Direct Mode Command: ContactCard Transmit Data

Definition at line 123 of file p3fw_dm.h.

4.3.1.17 #define P3FW_DM_CONTACTCARD_WARM_RESET 0x03

Direct Mode Command: ContactCard Warm Reset

Definition at line 118 of file p3fw_dm.h.

4.3.1.18 #define P3FW_DM_HAL 0xB0

Direct Mode Class: HAL

Definition at line 31 of file p3fw_dm.h.

4.3.1.19 #define P3FW_DM_HAL_APP_PROT_SET 0x05

Direct Mode Command: HAL Set Protocol

Definition at line 58 of file p3fw_dm.h.

4.3.1.20 #define P3FW_DM_HAL_EXEC_CMD 0x08

Direct Mode Command: HAL Execute Command

Definition at line 61 of file p3fw_dm.h.

4.3.1.21 #define P3FW_DM_HAL_GET_CFG 0x04

Direct Mode Command: HAL Get Configuration

Definition at line 57 of file p3fw_dm.h.

4.3.1.22 #define P3FW_DM_HAL_INIT 0x01

Direct Mode Command: HAL Init

Definition at line 54 of file p3fw_dm.h.

4.3.1.23 #define P3FW_DM_HAL_MFC_AUTH 0x07

Direct Mode Command: HAL MIFARE Auth

Definition at line 60 of file p3fw_dm.h.

4.3.1.24 #define P3FW_DM_HAL_MFC_AUTH_KEY 0x09

Direct Mode Command: HAL MIFARE Auth with KeyStore

Definition at line 62 of file p3fw_dm.h.

4.3.1.25 #define P3FW_DM_HAL_SET_CFG 0x03

Direct Mode Command: HAL Set Configuration

Definition at line 56 of file p3fw_dm.h.

4.3.1.26 #define P3FW_DM_HAL_WAIT 0x06

Direct Mode Command: HAL Wait

Definition at line 59 of file p3fw_dm.h.

4.3.1.27 #define P3FW_DM_HAL_XCHG 0x02

Direct Mode Command: HAL Exchange

Definition at line 55 of file p3fw_dm.h.

4.3.1.28 #define P3FW_DM_KSTOR 0x70

Direct Mode Class: Key Store

Definition at line 36 of file p3fw_dm.h.

4.3.1.29 #define P3FW_DM_KSTOR_CHG_KUC 0x0C

Direct Mode Command: KeyStore Change KUC

Definition at line 111 of file p3fw_dm.h.

4.3.1.30 #define P3FW_DM_KSTOR_FORMAT_KEY 0x02

Direct Mode Command: KeyStore Format Key

Definition at line 102 of file p3fw_dm.h.

4.3.1.31 #define P3FW_DM_KSTOR_GET_CFG_STR 0x0F

Direct Mode Command: KeyStore Get Configuration String

Definition at line 114 of file p3fw_dm.h.

4.3.1.32 #define P3FW_DM_KSTOR_GET_CONFIG 0x0B

Direct Mode Command: KeyStore Get Configuration

Definition at line 110 of file p3fw_dm.h.

4.3.1.33 #define P3FW_DM_KSTOR_GET_KEY 0x09

Direct Mode Command: KeyStore Get Key

Definition at line 108 of file p3fw_dm.h.

4.3.1.34 #define P3FW_DM_KSTOR_GET_KEY_ENTRY 0x08

Direct Mode Command: KeyStore Set Key Entry

Definition at line 107 of file p3fw_dm.h.

4.3.1.35 #define P3FW_DM_KSTOR_GET_KUC 0x0D

Direct Mode Command: KeyStore Get KUC

Definition at line 112 of file p3fw_dm.h.

4.3.1.36 #define P3FW_DM_KSTOR_INIT 0x01

Direct Mode Command: KeyStore Init

Definition at line 101 of file p3fw_dm.h.

4.3.1.37 #define P3FW_DM_KSTOR_SET_CFG_STR 0x0E

Direct Mode Command: KeyStore Set Configuration String

Definition at line 113 of file p3fw_dm.h.

4.3.1.38 #define P3FW_DM_KSTOR_SET_CONFIG 0x0A

Direct Mode Command: KeyStore Set Configuration

Definition at line 109 of file p3fw_dm.h.

4.3.1.39 #define P3FW_DM_KSTOR_SET_FULL_KEY 0x07

Direct Mode Command: KeyStore Set Full Key Entry

Definition at line 106 of file p3fw_dm.h.

4.3.1.40 #define P3FW_DM_KSTOR_SET_KEY 0x03

Direct Mode Command: KeyStore Set Key

Definition at line 103 of file p3fw_dm.h.

4.3.1.41 #define P3FW_DM_KSTOR_SET_KEY_POS 0x04

Direct Mode Command: KeyStore KeyPos

Definition at line 104 of file p3fw_dm.h.

4.3.1.42 #define P3FW_DM_KSTOR_SET_KUC 0x05

Direct Mode Command: KeyStore Set KUC

Definition at line 105 of file p3fw_dm.h.

4.3.1.43 #define P3FW_DM_L3 0xC0

Direct Mode Class: ISO14443-3

Definition at line 32 of file p3fw_dm.h.

4.3.1.44 #define P3FW_DM_L3_ACT_CARD 0x07

Direct Mode Command: ISO14443-3 Activate Card

Definition at line 70 of file p3fw_dm.h.

4.3.1.45 #define P3FW_DM_L3_ANTICOL 0x05

Direct Mode Command: ISO14443-3 Anticolision

Definition at line 68 of file p3fw_dm.h.

4.3.1.46 #define P3FW_DM_L3_GET_SER 0x09

Direct Mode Command: ISO14443-3 Get Serial

Definition at line 72 of file p3fw_dm.h.

4.3.1.47 #define P3FW_DM_L3_HLTA 0x04

Direct Mode Command: ISO14443-3 Halt

Definition at line 67 of file p3fw_dm.h.

4.3.1.48 #define P3FW_DM_L3_INIT 0x01

Direct Mode Command: ISO14443-3 Init

Definition at line 64 of file p3fw_dm.h.

4.3.1.49 #define P3FW_DM_L3_REQA 0x02

Direct Mode Command: ISO14443-3 ReqA

Definition at line 65 of file p3fw_dm.h.

4.3.1.50 #define P3FW_DM_L3_SELECT 0x06

Direct Mode Command: ISO14443-3 Select

Definition at line 69 of file p3fw_dm.h.

4.3.1.51 #define P3FW_DM_L3_WKUA 0x03

Direct Mode Command: ISO14443-3 WkuA

Definition at line 66 of file p3fw_dm.h.

4.3.1.52 #define P3FW_DM_L3_XCHG 0x08

Direct Mode Command: ISO14443-3 Exchange

Definition at line 71 of file p3fw_dm.h.

4.3.1.53 #define P3FW_DM_L4 0xE0

Direct Mode Class: ISO14443-4

Definition at line 34 of file p3fw_dm.h.

4.3.1.54 #define P3FW_DM_L4_DESELECT 0x04

Direct Mode Command: ISO14443-4 Deselect

Definition at line 83 of file p3fw_dm.h.

4.3.1.55 #define P3FW_DM_L4_GET_CFG 0x08

Direct Mode Command: ISO14443-4 Get Configuration

Definition at line 87 of file p3fw_dm.h.

4.3.1.56 #define P3FW_DM_L4_INIT 0x01

Direct Mode Command: ISO14443-4 Init

Definition at line 80 of file p3fw_dm.h.

4.3.1.57 #define P3FW_DM_L4_PRESENT_CHECK 0x05

Direct Mode Command: ISO14443-4 Present Check

Definition at line 84 of file p3fw_dm.h.

4.3.1.58 #define P3FW_DM_L4_RESET_PROTO 0x03

Direct Mode Command: ISO14443-4 Reset Protocol

Definition at line 82 of file p3fw_dm.h.

4.3.1.59 #define P3FW_DM_L4_SET_CFG 0x07

Direct Mode Command: ISO14443-4 Set Configuration

Definition at line 86 of file p3fw_dm.h.

4.3.1.60 #define P3FW_DM_L4_SET_PROTO 0x02

Direct Mode Command: ISO14443-4 Set Protocol

Definition at line 81 of file p3fw_dm.h.

4.3.1.61 #define P3FW_DM_L4_XCHG 0x06

Direct Mode Command: ISO14443-4 Exchange

Definition at line 85 of file p3fw_dm.h.

4.3.1.62 #define P3FW_DM_L4A 0xD0

Direct Mode Class: ISO14443-4A

Definition at line 33 of file p3fw_dm.h.

4.3.1.63 #define P3FW_DM_L4A_ACT_CARD 0x04

Direct Mode Command: ISO14443-4A Activate Card

Definition at line 77 of file p3fw_dm.h.

4.3.1.64 #define P3FW_DM_L4A_GET_PROTO_PARM 0x05

Direct Mode Command: ISO14443-4A Get Protocol Parameters

Definition at line 78 of file p3fw_dm.h.

4.3.1.65 #define P3FW_DM_L4A_INIT 0x01

Direct Mode Command: ISO14443-4A Init

Definition at line 74 of file p3fw_dm.h.

4.3.1.66 #define P3FW_DM_L4A_PPS 0x03

Direct Mode Command: ISO14443-4A PPS

Definition at line 76 of file p3fw_dm.h.

4.3.1.67 #define P3FW_DM_L4A_RATS 0x02

Direct Mode Command: ISO14443-4A RATS

Definition at line 75 of file p3fw_dm.h.

4.3.1.68 #define P3FW_DM_RO_0xA0

Direct Mode Class: Reader Operations

Definition at line 30 of file p3fw_dm.h.

4.3.1.69 #define P3FW_DM_RO_CONF_OVER_0x04

Direct Mode Command: Configuration Overwrite

Definition at line 42 of file p3fw_dm.h.

4.3.1.70 #define P3FW_DM_RO_FIELD_OFF_0x0A

Direct Mode Command: Field Off

Definition at line 49 of file p3fw_dm.h.

4.3.1.71 #define P3FW_DM_RO_FIELD_ON_0x09

Direct Mode Command: Field On

Definition at line 48 of file p3fw_dm.h.

4.3.1.72 #define P3FW_DM_RO_FIELD_RESET_0x0B

Direct Mode Command: Field Reset

Definition at line 50 of file p3fw_dm.h.

4.3.1.73 #define P3FW_DM_RO_GET_CONF_0x0C

Direct Mode Command: Get Configuration

Definition at line 44 of file p3fw_dm.h.

4.3.1.74 #define P3FW_DM_RO_GET_STATUS_0x06

Direct Mode Command: Get Status

Definition at line 45 of file p3fw_dm.h.

4.3.1.75 #define P3FW_DM_RO_LEDS_OFF_0x01

Direct Mode Command: Led Off

Definition at line 39 of file p3fw_dm.h.

4.3.1.76 #define P3FW_DM_RO_LEDS_ON 0x02

Direct Mode Command: Led On

Definition at line 40 of file p3fw_dm.h.

4.3.1.77 #define P3FW_DM_RO_READ_REG 0x07

Direct Mode Command: Read Register

Definition at line 46 of file p3fw_dm.h.

4.3.1.78 #define P3FW_DM_RO_RESET 0x03

Direct Mode Command: Reset

Definition at line 41 of file p3fw_dm.h.

4.3.1.79 #define P3FW_DM_RO_SET_CONF 0x05

Direct Mode Command: Set Configuration

Definition at line 43 of file p3fw_dm.h.

4.3.1.80 #define P3FW_DM_RO_SET_PCSC_MODE 0x0D

Direct Mode Command: Set PCSC Mode

Definition at line 51 of file p3fw_dm.h.

4.3.1.81 #define P3FW_DM_RO_TEST_MODE 0x0E

Direct Mode Command: Test Mode

Definition at line 52 of file p3fw_dm.h.

4.3.1.82 #define P3FW_DM_RO_WRITE_REG 0x08

Direct Mode Command: Write Register

Definition at line 47 of file p3fw_dm.h.

4.3.1.83 #define P3FW_DM_XCHG 0xF0

Direct Mode Class: Exchange

Definition at line 35 of file p3fw_dm.h.

4.3.1.84 #define P3FW_DM_XCHG_INIT 0x07

Direct Mode Command: MIFARE Init

Definition at line 95 of file p3fw_dm.h.

4.3.1.85 #define P3FW_DM_XCHG_L3 0x01

Direct Mode Command: MIFARE Exchange L3

Definition at line 89 of file p3fw_dm.h.

4.3.1.86 #define P3FW_DM_XCHG_L4 0x02

Direct Mode Command: MIFARE Exchange L4

Definition at line 90 of file p3fw_dm.h.

4.3.1.87 #define P3FW_DM_XCHG_MFC_AUTH 0x05

Direct Mode Command: MIFARE Auth

Definition at line 93 of file p3fw_dm.h.

4.3.1.88 #define P3FW_DM_XCHG_MFC_AUTH_KEY 0x06

Direct Mode Command: MIFARE Auth with KeyStore

Definition at line 94 of file p3fw_dm.h.

4.3.1.89 #define P3FW_DM_XCHG_PC 0x03

Direct Mode Command: MIFARE Exchange PC

Definition at line 91 of file p3fw_dm.h.

4.3.1.90 #define P3FW_DM_XCHG_RAW 0x04

Direct Mode Command: MIFARE Exchange RAW

Definition at line 92 of file p3fw_dm.h.

4.3.2 Function Documentation**4.3.2.1 void p3fw_dm_cid ()**

Direct Mode CID Management Functions.

Definition at line 26 of file p3fw_dm_cid.c.

4.3.2.2 void p3fw_dm_contact_card ()

Direct Mode Contact Card Functions.

Definition at line 25 of file p3fw_dm_cc.c.

4.3.2.3 uint8_t p3fw_dm_get_class ()

Extracts class from the frame.

Returns

class

Definition at line 70 of file p3fw_dm.c.

4.3.2.4 uint8_t* p3fw_dm_get_input_payload ()

Gets input payload buffer.

Returns

pointer to buffer

Definition at line 86 of file p3fw_dm.c.

4.3.2.5 uint16_t p3fw_dm_get_input_payload_length ()

Gets input length.

Returns

length

Definition at line 80 of file p3fw_dm.c.

4.3.2.6 uint8_t* p3fw_dm_get_input_pointer_uint8_t (uint16_t wDmByteIndex)

Definition at line 111 of file p3fw_dm.c.

4.3.2.7 uint16_t p3fw_dm_get_input_uint16_t (uint16_t wDmByteIndex)

Definition at line 116 of file p3fw_dm.c.

4.3.2.8 uint32_t p3fw_dm_get_input_uint32_t (uint16_t wDmByteIndex)

Definition at line 122 of file p3fw_dm.c.

4.3.2.9 uint8_t p3fw_dm_get_input_uint8_t (uint16_t wDmByteIndex)

Definition at line 106 of file p3fw_dm.c.

4.3.2.10 uint8_t p3fw_dm_get_instruction ()

Extracts instruction from the frame.

Returns

instruction

Definition at line 75 of file p3fw_dm.c.

4.3.2.11 uint8_t* p3fw_dm_get_output_payload ()

Gets output payload buffer.

Returns

pointer to buffer

Definition at line 91 of file p3fw_dm.c.

4.3.2.12 uint8_t* p3fw_dm_get_output_pointer_uint8_t (uint16_t wDmByteIndex)

Definition at line 96 of file p3fw_dm.c.

4.3.2.13 uint8_t p3fw_dm_get_slot_index ()

Extracts slot index from the frame.

Returns

slot index

Definition at line 101 of file p3fw_dm.c.

4.3.2.14 void p3fw_dm_hal ()

Direct Mode Hardware Abstraction Layer Functions.

Definition at line 33 of file p3fw_dm_hal.c.

4.3.2.15 void p3fw_dm_keystore ()

Direct Mode Keystore Management Functions.

Definition at line 38 of file p3fw_dm_keystore.c.

4.3.2.16 void p3fw_dm_l3 ()

Direct Mode ISO14443-3 Functions.

Definition at line 32 of file p3fw_dm_l3.c.

4.3.2.17 void p3fw_dm_l4 ()

Direct Mode ISO14443-4 Functions.

Definition at line 31 of file p3fw_dm_l4.c.

4.3.2.18 void p3fw_dm_l4a ()

Direct Mode ISO14443-4A Functions.

Definition at line 28 of file p3fw_dm_l4a.c.

4.3.2.19 void p3fw_dm_ro ()

Direct Mode Reader Operations (also available through SCardControl/CCID Escape Interface)

Definition at line 39 of file p3fw_dm_ro.c.

4.3.2.20 void p3fw_dm_send_frame (phStatus_t status, uint16_t wFrameLength)

Sends frame.

Parameters

in	<i>status</i>	NXPRdLib Status Code
in	<i>wFrameLength</i>	Frame length in bytes

Definition at line 130 of file p3fw_dm.c.

4.3.2.21 void p3fw_dm_xchg ()

Direct Mode Exchange Functions.

Definition at line 30 of file p3fw_dm_xchg.c.

4.3.2.22 void p3fw_dm_xfer ()

Direct Mode Transfer Mode.

Definition at line 23 of file p3fw_dm.c.

4.4 Firmware

Data Structures

- struct `p3fw_hal_buffer`
Every HAL requires buffer for sending and receiving.
- struct `_p3fw_cl_slot`
Stores state of a contact less slot.
- struct `_p3fw_cc_slot`
- struct `p3fw_data`
Stores all reader global settings and variables.
- struct `p3_fw_ccid_exec`
Job parameters for.

Defines

- #define `P3FW_MAX_CCID_BUFFER_LEN` 271
- #define `P3FW_MAX_CONTACTLESS_SLOTS` 14
- #define `P3FW_MAX_CONTACT_SLOTS` 1
- #define `P3FW_CONTACT_SLOTS_MASK` 0x80
- #define `P3FW_MAX_UID_A_LEN` 10
- #define `P3FW_MAX_UID_B_LEN` 4
- #define `P3FW_MAX_ATQB_LEN` 14
- #define `P3FW_MAX_HAL_TX_BUFFER` 256
- #define `P3FW_MAX_HAL_RX_BUFFER` 256
- #define `P3FW_PCSC_FSDI` 5
- #define `P3FW_MAX_ATS_LEN` 64
- #define `P3FW_MAX_KEY_LEN` 6
- #define `P3FW_MAJOR` 0x02U
- #define `P3FW_MINOR` 0x02U
- #define `P3FW_BUILD` 0x07U
- #define `P3FW_CFG_MAX_SLOTS` 0xCA000001U
Configuration options.
- #define `P3FW_CFG_CONT_TIMING` 0xCB000001U
- #define `P3FW_CFG_GET_CONT_TIMING` 0xCC000001U
- #define `P3FW_CFG_SET_DIP_SWITCHES` 0xCD000001U
- #define `P3FW_CFG_BOOTLOADER_VERSION` 0xEF000001U
- #define `P3FW_CFG_BOOTLOADER_ACTIVE` 0xEF000002U
- #define `P3FW_ERR_AND_NFO_LOOP_DONE` 0x00000001U
Error and notifications codes for err_and_nfo_mode.

- `#define P3FW_ERR_AND_NFO_LOOP_UNKNOWN_ERROR 0x00000002U`
- `#define P3FW_ERR_AND_NFO_LOOP_CHIP_NOT_DETECTED 0x00000003U`
- `#define P3FW_ERR_AND_NFO_LOOP_COM_ERROR 0x00000004U`
- `#define P3FW_ERR_AND_NFO_LOOP_HAL_CAN_NOT_BE_SET 0x00000005U`
- `#define P3FW_ERR_AND_NFO_LOOP_HW_ERROR 0x00000006U`
- `#define P3FW_ERR_AND_NFO_LOOP_OS_ERROR 0x00000007U`
- `#define P3FW_ERR_AND_NFO_LOOP_ERASE_FAILED 0x00000008U`
- `#define P3FW_ERR_AND_NFO_LOOP_FLASH_FAILED 0x00000009U`
- `#define P3FW_ERR_AND_NFO_LOOP_BFL_ERROR 0x0000000AU`
- `#define P3FW_ERR_AND_NFO_LOOP_USB_ERROR 0x0000000BU`
- `#define P3FW_KEYSTORE_NUM_KEYS 0x05`
- `#define P3FW_KEYSTORE_NUM_VERS 0x05`
- `#define P3FW_USB_INT_IN_EP 0x81`
- `#define P3FW_USB_BULK_OUT_EP 0x05`
- `#define P3FW_USB_BULK_IN_EP 0x82`
- `#define P3FW_PCSC_PROTO_PARAMS_LEN_T0 0x05`
- `#define P3FW_PCSC_PROTO_PARAMS_LEN_T1 0x07`
- `#define P3FW_CHECK_STATUS(status, error_code)`

Typedefs

- `typedef struct _p3fw_cl_slot p3fw_cl_slot`
Stores state of a contact less slot.
- `typedef struct _p3fw_cc_slot p3fw_cc_slot`

Enumerations

- `enum p3fw_pcsc_mode { P3FW_PCSC_MODE_NORMAL, P3FW_PCSC_MODE_DIRECT }`
- `enum p3fw_slot_types {`
`P3FW_SLOT_EMPTY, P3FW_SLOT_SAM, P3FW_SLOT_ISO14443A_CARD,`
`P3FW_SLOT_ISO14443B_CARD,`
`P3FW_SLOT_ISO14444_CARD }`
Slot is occupied with card type of.
- `enum p3fw_bal_configuration { P3FW_BAL_UART, P3FW_BAL_SPI, P3FW_BAL_I2C }`
Interfaces NXPRdLib BAL can use.
- `enum p3fw_sam { P3FW_SAM_NO, P3FW_SAM_NON_X, P3FW_SAM_IN_X }`
How Pegoda uses the SAM.

- enum `p3fw_mode` { `P3FW_MODE_PCSC`, `P3FW_MODE_DEMO`, `P3FW_MODE_CFG_OVERWRITE`, `P3FW_MODE_ACTIVATE_BOOTLOADER` }

Available reader modes.

- enum `p3fw_external_interface` { `P3FW_EXT_IF_USB`, `P3FW_EXT_IF_RS232`, `P3FW_EXT_IF_RS485`, `P3FW_EXT_IF_ETHERNET` }

Interfaces Pegoda can use.

- enum `p3fw_ic` { `P3FW_IC_RC523` = 0x01U, `P3FW_IC_SAM` = 0x02U, `P3FW_IC_RC663` = 0x03U }

Reader ICs list.

- enum `p3fw_pcsc_protocol` { `P3FW_PCSC_PROTOCOL_T0` = 0x00U, `P3FW_PCSC_PROTOCOL_T1` = 0x01U, `P3FW_PCSC_PROTOCOL_RAW` = 0x02U, `P3FW_PCSC_PROTOCOL_UNKNOWN` = 0xFFU }

Functions

- void `p3fw_invoke_err_and_nfo_mode` (uint32_t dwErrorCode)
Invokes the error and information loop.
- void `p3fw_task_demo_mode` (void *param)
Task for self contained demo mode.
- void `p3fw_task_poll_and_activate` (void *param)
Task polling and activation of cards.
- void `p3fw_task_ccid_execute` (void *param)
Task for bottom half CCID handling.
- void `p3fw_flash_erase_config` (void)
Erases all configuration options saved on flash.
- phStatus_t `p3fw_flash_read_serial` (uint32_t *pSerNum)
Reads the LPC unique four byte serial number.
- phStatus_t `p3fw_flash_get_config` (uint32_t dwCfgId, uint8_t *pBuff)
Returns a configuration value designated with dwCfgId.
- phStatus_t `p3fw_flash_set_config` (uint32_t dwCfgId, uint8_t *pBuff)
Sets the configuration value designated with dwCfgId.
- phStatus_t `p3fw_slots_init` (void)
Initializes the basic BLF structures and slots data.
- void `p3fw_ext_if_init_usb` (void)

Initialize the USB external interface.

- void [p3fw_slot_add_l3a_card](#) (uint8_t bSCLlotIndex, uint8_t bSak)
Adds a ISO14443 L3A card to the slot /c bCLSlotIndex.
- void [p3fw_slot_add_l3b_card](#) (uint8_t bCLSlotIndex)
Adds a ISO14443 L3B card to the slot /c bCLSlotIndex.
- void [p3fw_slot_add_l4_card](#) (uint8_t bCLSlotIndex, uint8_t bSak)
Adds a ISO14443 L4 card to the slot /c bCLSlotIndex.
- void [p3fw_slot_remove_cl_card](#) (uint8_t bCLSlotIndex)
Removes card from slot /c bCLSlotIndex.
- phStatus_t [p3fw_slot_get_atr](#) (uint8_t bCCIDSlotIndex, uint8_t *pbAtrBuffer, uint8_t *pbMaxLength)
Returns ATR for a slot /c bCCIDSlotIndex.
- void [p3fw_slot_reset_all_slots](#) (uint8_t contact_card)
Resets all slots to init state.
- void [p3fw_timing_init](#) (void)
Inits the timing interface.
- void [p3fw_timing_start](#) ()
Resets and starts the timer.
- uint32_t [p3fw_timing_stop](#) ()
Stops the timer and return the elapsed value.

Variables

- [p3fw_data](#) p3fw

4.4.1 Define Documentation

4.4.1.1 #define P3FW_BUILD 0x07U

Version Build number

Definition at line 74 of file p3fw_fw.h.

4.4.1.2 #define P3FW_CFG_BOOTLOADER_ACTIVE 0xEF000002U

Set Bootloader activation

Definition at line 84 of file p3fw_fw.h.

4.4.1.3 #define P3FW_CFG_BOOTLOADER_VERSION 0xEF000001U

Get Bootloader Version

Definition at line 83 of file p3fw_fw.h.

4.4.1.4 #define P3FW_CFG_CONT_TIMING 0xCB000001U

Timing mode for contact cards

Definition at line 80 of file p3fw_fw.h.

4.4.1.5 #define P3FW_CFG_GET_CONT_TIMING 0xCC000001U

Get last timing value

Definition at line 81 of file p3fw_fw.h.

4.4.1.6 #define P3FW_CFG_MAX_SLOTS 0xCA000001U

Configuration options.

Number of slots reported by the reader

Definition at line 79 of file p3fw_fw.h.

4.4.1.7 #define P3FW_CFG_SET_DIP_SWITCHES 0xCD000001U

Set DIP switches with configuration - used for testing only

Definition at line 82 of file p3fw_fw.h.

4.4.1.8 #define P3FW_CHECK_STATUS(status, error_code)

Value:

```
if (PH_ERR_SUCCESS != status) {
    \
    _DBG_L("status: "); _DBH16L(status); _DBG_L(" ");
    \
    _DBG_L("file   : "); _DBGHL((uint8_t *)__FILE__, sizeof(__FILE__)); \
    _DBG_L("line   : "); _DBH32L(__LINE__); _DBG_L(" ");
    \
    p3fw_invoke_err_and_nfo_mode (error_code); }
```

Definition at line 112 of file p3fw_fw.h.

4.4.1.9 #define P3FW_CONTACT_SLOTS_MASK 0x80

Definition at line 56 of file p3fw_fw.h.

4.4.1.10 #define P3FW_ERR_AND_NFO_LOOP_BFL_ERROR 0x0000000AU

BFL Error

Definition at line 98 of file p3fw_fw.h.

4.4.1.11 #define P3FW_ERR_AND_NFO_LOOP_CHIP_NOT_DETECTED 0x00000003U

Reader chip not detected

Definition at line 91 of file p3fw_fw.h.

4.4.1.12 #define P3FW_ERR_AND_NFO_LOOP_COM_ERROR 0x00000004U

Communication error

Definition at line 92 of file p3fw_fw.h.

4.4.1.13 #define P3FW_ERR_AND_NFO_LOOP_DONE 0x00000001U

Error and notifications codes for err_and_nfo_mode.

Operation completed

Definition at line 89 of file p3fw_fw.h.

4.4.1.14 #define P3FW_ERR_AND_NFO_LOOP_ERASE_FAILED 0x00000008U

Erase failed

Definition at line 96 of file p3fw_fw.h.

4.4.1.15 #define P3FW_ERR_AND_NFO_LOOP_FLASH_FAILED 0x00000009U

Flash failed

Definition at line 97 of file p3fw_fw.h.

4.4.1.16 #define P3FW_ERR_AND_NFO_LOOP_HAL_CAN_NOT_BE_SET 0x00000005U

Error in HAL layer

Definition at line 93 of file p3fw_fw.h.

4.4.1.17 #define P3FW_ERR_AND_NFO_LOOP_HW_ERROR 0x00000006U

Hardware error

Definition at line 94 of file p3fw_fw.h.

4.4.1.18 #define P3FW_ERR_AND_NFO_LOOP_OS_ERROR 0x00000007U

Operating system error

Definition at line 95 of file p3fw_fw.h.

4.4.1.19 #define P3FW_ERR_AND_NFO_LOOP_UNKNOWN_ERROR 0x00000002U

Unknown error

Definition at line 90 of file p3fw_fw.h.

4.4.1.20 #define P3FW_ERR_AND_NFO_LOOP_USB_ERROR 0x0000000BU

USB Communication Error

Definition at line 99 of file p3fw_fw.h.

4.4.1.21 #define P3FW_KEYSTORE_NUM_KEYS 0x05

Number of key in the key store

Definition at line 101 of file p3fw_fw.h.

4.4.1.22 #define P3FW_KEYSTORE_NUM_VERS 0x05

Number of versions of keys

Definition at line 102 of file p3fw_fw.h.

4.4.1.23 #define P3FW_MAJOR 0x02U

Firmware version Version Major number

Definition at line 72 of file p3fw_fw.h.

4.4.1.24 #define P3FW_MAX_ATQB_LEN 14

Maximum ATQB length

Definition at line 61 of file p3fw_fw.h.

4.4.1.25 #define P3FW_MAX_ATS_LEN 64

Maximum ATS length - depends on FSDI

Definition at line 67 of file p3fw_fw.h.

4.4.1.26 #define P3FW_MAX_CCID_BUFFER_LEN 271

CCID buffer length

Definition at line 50 of file p3fw_fw.h.

4.4.1.27 #define P3FW_MAX_CONTACT_SLOTS 1

Maximum number of available * contact slots

Definition at line 54 of file p3fw_fw.h.

4.4.1.28 #define P3FW_MAX_CONTACTLESS_SLOTS 14

Maximum number of available contact-less slots

Definition at line 52 of file p3fw_fw.h.

4.4.1.29 #define P3FW_MAX_HAL_RX_BUFFER 256

Definition at line 64 of file p3fw_fw.h.

4.4.1.30 #define P3FW_MAX_HAL_TX_BUFFER 256

Definition at line 63 of file p3fw_fw.h.

4.4.1.31 #define P3FW_MAX_KEY_LEN 6

Maximum MIFARE Classic Key

Definition at line 68 of file p3fw_fw.h.

4.4.1.32 #define P3FW_MAX_UID_A_LEN 10

Maximum UID length

Definition at line 58 of file p3fw_fw.h.

4.4.1.33 #define P3FW_MAX_UID_B_LEN 4

Maximum UID length

Definition at line 59 of file p3fw_fw.h.

4.4.1.34 **#define P3FW_MINOR 0x02U**

Version Minor number

Definition at line 73 of file p3fw_fw.h.

4.4.1.35 **#define P3FW_PCSC_FSDI 5**

Maximum FSDI => ATS length

Definition at line 66 of file p3fw_fw.h.

4.4.1.36 **#define P3FW_PCSC_PROTO_PARAMS_LEN_T0 0x05**

Length of parameters for T=0

Definition at line 108 of file p3fw_fw.h.

4.4.1.37 **#define P3FW_PCSC_PROTO_PARAMS_LEN_T1 0x07**

Length of parameters for T=1

Definition at line 109 of file p3fw_fw.h.

4.4.1.38 **#define P3FW_USB_BULK_IN_EP 0x82**

Definition at line 106 of file p3fw_fw.h.

4.4.1.39 **#define P3FW_USB_BULK_OUT_EP 0x05**

Definition at line 105 of file p3fw_fw.h.

4.4.1.40 **#define P3FW_USB_INT_IN_EP 0x81**

Definition at line 104 of file p3fw_fw.h.

4.4.2 Typedef Documentation

4.4.2.1 **typedef struct _p3fw_cc_slot p3fw_cc_slot**

4.4.2.2 **typedef struct _p3fw_cl_slot p3fw_cl_slot**

Stores state of a contact less slot.

4.4.3 Enumeration Type Documentation

4.4.3.1 enum p3fw_bal_configuration

Interfaces NXPRdLib BAL can use.

Enumerator:

P3FW_BAL_UART UART Interface

P3FW_BAL_SPI SPI Interface

P3FW_BAL_I2C I2C Interface

Definition at line 139 of file p3fw_fw.h.

4.4.3.2 enum p3fw_external_interface

Interfaces Pegoda can use.

Enumerator:

P3FW_EXT_IF_USB USB Interface

P3FW_EXT_IF_RS232 RS232 Interface

P3FW_EXT_IF_RS485 RS485 Interface

P3FW_EXT_IF_ETHERNET Ethernet Interface

Definition at line 170 of file p3fw_fw.h.

4.4.3.3 enum p3fw_ic

Reader ICs list.

Enumerator:

P3FW_IC_RC523 RC523

P3FW_IC_SAM SAM or RX852

P3FW_IC_RC663 RC633

Definition at line 181 of file p3fw_fw.h.

4.4.3.4 enum p3fw_mode

Available reader modes.

Enumerator:

P3FW_MODE_PCSC PCSC Mode

P3FW_MODE_DEMO Demo Mode

P3FW_MODE_CFG_OVERWRITE Overwrite configuration

P3FW_MODE_ACTIVATE_BOOTLOADER Activate Bootloader

Definition at line 159 of file p3fw_fw.h.

4.4.3.5 enum p3fw_pcsc_mode

Enumerator:

P3FW_PCSC_MODE_NORMAL

P3FW_PCSC_MODE_DIRECT

Definition at line 118 of file p3fw_fw.h.

4.4.3.6 enum p3fw_pcsc_protocol

Enumerator:

P3FW_PCSC_PROTOCOL_T0 T=0

P3FW_PCSC_PROTOCOL_T1 T=1

P3FW_PCSC_PROTOCOL_RAW T=RAW

P3FW_PCSC_PROTOCOL_UNKNOWN T=UNKNOWN

Definition at line 188 of file p3fw_fw.h.

4.4.3.7 enum p3fw_sam

How Pegoda uses the SAM.

Enumerator:

P3FW_SAM_NO No SAM

P3FW_SAM_NON_X SAM in S mode

P3FW_SAM_IN_X SAM in X mode

Definition at line 149 of file p3fw_fw.h.

4.4.3.8 enum p3fw_slot_types

Slot is occupied with card type of.

Enumerator:

P3FW_SLOT_EMPTY Slot is empty

P3FW_SLOT_SAM Card is SAM

P3FW_SLOT_ISO1444L3A_CARD Card is ISO14443-3A

P3FW_SLOT_ISO1444L3B_CARD Card is ISO14443-3B

P3FW_SLOT_ISO1444L4_CARD Card is ISO14443-4

Definition at line 127 of file p3fw_fw.h.

4.4.4 Function Documentation

4.4.4.1 void p3fw_ext_if_init_usb (void)

Initialize the USB external interface.

Definition at line 177 of file p3fw_ext_if_usb.c.

4.4.4.2 void p3fw_flash_erase_config (void)

Erases all configuration options saved on flash.

Returns

void

Definition at line 54 of file p3fw_flash.c.

4.4.4.3 phStatus_t p3fw_flash_get_config (uint32_t dwCfgId, uint8_t * pBuff)

Returns a configuration value designated with dwCfgId.

Returns

Status code

Return values

#PH_ERR_SUCCESS	Operation successful.
#PH_ERR_INTERNAL_ERROR	on error

Parameters

in	<i>dwCfgId</i>	Configuration id
out	<i>pBuff</i>	Buffer to save value to

Definition at line 86 of file p3fw_flash.c.

4.4.4.4 phStatus_t p3fw_flash_read_serial (uint32_t * pSerNum)

Reads the LPC unique four byte serial number.

Returns

Status code

Return values

<i>#PH_ERR_SUCCESS</i>	Operation successful.
<i>#PH_ERR_INTERNAL_ERROR</i>	on error

Parameters

out	<i>pSerNum</i>	Buffer to save the serial number
-----	----------------	----------------------------------

Definition at line 63 of file p3fw_flash.c.

4.4.4.5 phStatus_t p3fw_flash_set_config (uint32_t dwCfgId, uint8_t * pBuff)

Sets the configuration value designated with dwCfgId.

Returns

Status code

Return values

<i>#PH_ERR_SUCCESS</i>	Operation successful.
<i>#PH_ERR_INTERNAL_ERROR</i>	on error

Parameters

in	<i>dwCfgId</i>	Configuration id
in	<i>pBuff</i>	Buffer with value

Definition at line 136 of file p3fw_flash.c.

4.4.4.6 void p3fw_invoke_err_and_nfo_mode (uint32_t dwErrorCode)

Invokes the error and information loop.

Returns

void

Parameters

in	<i>dwError-Code</i>	The Error or Information to report
----	---------------------	------------------------------------

Definition at line 22 of file p3fw_error_mode.c.

4.4.4.7 void p3fw_slot_add_l3a_card (uint8_t bSCLlotIndex, uint8_t bSak)

Adds a ISO14443 L3A card to the slot /c bCLSlotIndex.

Parameters

in	<i>bSCLlotIndex</i>	Slot to use
in	<i>bSak</i>	Card SAK number

Definition at line 357 of file p3fw_slots.c.

4.4.4.8 void p3fw_slot_add_l3b_card (uint8_t bCLSlotIndex)

Adds a ISO14443 L3B card to the slot /c bCLSlotIndex.

Parameters

in	<i>bCLSlotIndex</i>	Slot to use
----	---------------------	-------------

Definition at line 377 of file p3fw_slots.c.

4.4.4.9 void p3fw_slot_add_l4_card (uint8_t bCLSlotIndex, uint8_t bSak)

Adds a ISO14443 L4 card to the slot /c bCLSlotIndex.

Parameters

in	<i>bCLSlotIndex</i>	Slot to use
in	<i>bSak</i>	Card SAK number

Definition at line 396 of file p3fw_slots.c.

4.4.4.10 phStatus_t p3fw_slot_get_atr (uint8_t bCCIDSlotIndex, uint8_t * pbAtrBuffer, uint8_t * pbMaxLength)

Returns ATR for a slot /c bCCIDSlotIndex.

Returns

Status code

Return values

<i>#PH_ERR_-SUCCESS</i>	Operation successful.
-------------------------	-----------------------

Parameters

in	<i>bCCID-SlotIndex</i>	Slot to use
out	<i>pbAtrBuffer</i>	Buffer for ATR
in, out	<i>pb-MaxLength</i>	Buffer length/ATR length

Definition at line 499 of file p3fw_slots.c.

4.4.4.11 void p3fw_slot_remove_cl_card (uint8_t bCLSlotIndex)

Removes card from slot /c bCLSlotIndex.

Parameters

in	<i>bCLSlotIndex</i>	Slot to use
----	---------------------	-------------

Definition at line 447 of file p3fw_slots.c.

4.4.4.12 void p3fw_slot_reset_all_slots (uint8_t contact_card)

Resets all slots to init state.

Definition at line 529 of file p3fw_slots.c.

4.4.4.13 phStatus_t p3fw_slots_init (void)

Initializes the basic BLF structures and slots data.

Returns

Status code

Return values

<i>#PH_ERR_-SUCCESS</i>	Operation successful.
<i>phStatus_t</i>	error code

Definition at line 31 of file p3fw_slots.c.

4.4.4.14 void p3fw_task_ccid_execute (void * *param*)

Task for bottom half CCID handling.

Returns

void

Parameters

<i>in</i>	<i>param</i>	Requierd by FreeRTOS - not used
-----------	--------------	---------------------------------

Definition at line 22 of file p3fw_ccid.c.

4.4.4.15 void p3fw_task_demo_mode (void * *param*)

Task for self contained demo mode.

Returns

void

Parameters

<i>in</i>	<i>param</i>	Requierd by FreeRTOS - not used
-----------	--------------	---------------------------------

Definition at line 46 of file p3fw_demo.c.

4.4.4.16 void p3fw_task_poll_and_activate (void * *param*)

Task polling and activation of cards.

Returns

void

Parameters

<i>in</i>	<i>param</i>	Requierd by FreeRTOS - not used
-----------	--------------	---------------------------------

Definition at line 32 of file p3fw_poll.c.

4.4.4.17 void p3fw_timing_init (void)

Inits the timing interface.

Definition at line 24 of file p3fw_timing.c.

4.4.4.18 void p3fw_timing_start ()

Resets and starts the timer.

Definition at line 34 of file p3fw_timing.c.

4.4.4.19 uint32_t p3fw_timing_stop ()

Stops the timer and return the elapsed value.

Returns

elapsed time

Definition at line 40 of file p3fw_timing.c.

4.4.5 Variable Documentation

4.4.5.1 p3fw_data p3fw

Definition at line 45 of file p3fw.c.

4.5 Initialization

Defines

- #define [P3FW_HW_SIGNAL_ANTENNA](#) 0x01
Signals.
- #define [P3FW_HW_SIGNAL_BEEPER](#) 0x08
- #define [P3FW_HW_SIGNAL_YELLOW_2](#) 0x10
- #define [P3FW_HW_SIGNAL_YELLOW_3](#) 0x20
- #define [P3FW_HW_SIGNAL_YELLOW_4](#) 0x40

Functions

- void [p3fw_hw_init](#) (void)
Initialize NXP LPC Hardware.
- void [p3fw_hw_ctrl_signal](#) (uint8_t bSignal, uint8_t bStatus)
Turn the blue Pegoda 2 antenna on or off.

4.5.1 Define Documentation

4.5.1.1 `#define P3FW_HW_SIGNAL_ANTENNA 0x01`

Signals.

Antenna

Definition at line 34 of file p3fw_hw.h.

4.5.1.2 `#define P3FW_HW_SIGNAL_BEEPER 0x08`

Beeper

Definition at line 35 of file p3fw_hw.h.

4.5.1.3 `#define P3FW_HW_SIGNAL_YELLOW_2 0x10`

Yellow LED 2

Definition at line 36 of file p3fw_hw.h.

4.5.1.4 `#define P3FW_HW_SIGNAL_YELLOW_3 0x20`

Yellow LED 3

Definition at line 37 of file p3fw_hw.h.

4.5.1.5 `#define P3FW_HW_SIGNAL_YELLOW_4 0x40`

Yellow LED 4

Definition at line 38 of file p3fw_hw.h.

4.5.2 Function Documentation

4.5.2.1 `void p3fw_hw_ctrl_signal (uint8_t bSignal, uint8_t bStatus)`

Turn the blue Pegoda 2 antenna on or off.

Returns

void

Definition at line 114 of file p3fw_bsp.c.

4.5.2.2 `void p3fw_hw_init (void)`

Initialize NXP LPC Hardware.

Returns

void

Definition at line 31 of file p3fw_bsp.c.

4.6 PCSCs

Functions

- phStatus_t [p3fw_pcsc_std_ext](#) (uint8_t bCLSlotIndex)
Checks for and execute a PCSC Standard Extensions.
- phStatus_t [p3fw_pcsc_send_apdu](#) (uint8_t bSW1, uint8_t bSW2, uint16_t wPayloadLength)
Sends a replay APDU with /c bSW1 and /c bSW2 as SW 1 and SW 2 and payload of length /c wPayloadLength.
- phStatus_t [p3fw_pcsc_std_ext_do_auth](#) (uint8_t bCLSlotIndex, uint8_t bBlockNumber, uint8_t bKeyType)
Authenticates a MIFARE Classic card with key and settings from previous PCSC Standard Extension commands.
- phStatus_t [p3fw_pcsc_std_ext_prepare_l3_card](#) (uint8_t bCLSlotIndex)
Halts and reactivates a L3 card so that it is always in correct state to receive PCSC Standard Extension commands.

4.6.1 Function Documentation

4.6.1.1 phStatus_t p3fw_pcsc_send_apdu (uint8_t bSW1, uint8_t bSW2, uint16_t wPayloadLength)

Sends a replay APDU with /c bSW1 and /c bSW2 as SW 1 and SW 2 and payload of length /c wPayloadLength.

Returns

Status code

Return values

#PH_ERR_SUCCESS	Operation successful.
-----------------	-----------------------

Parameters

in	bSW1	Replay APDUs SW1 byte
in	bSW2	Replay APDUs SW2 byte

in	<i>wPayloadLength</i>	Replay APDUs SW1 byte
----	-----------------------	-----------------------

Definition at line 23 of file p3fw_pcsc.c.

4.6.1.2 phStatus_t p3fw_pcsc_std_ext (uint8_t bCLSlotIndex)

Checks for and execute a PCSC Standard Extensions.

Returns

Status code

Return values

<i>#PH_ERR_SUCCESS</i>	Operation successful.
------------------------	-----------------------

Parameters

in	<i>bCLSlotIndex</i>	Slot index
----	---------------------	------------

Definition at line 30 of file p3fw_pcsc_mem_cards.c.

4.6.1.3 phStatus_t p3fw_pcsc_std_ext_do_auth (uint8_t bCLSlotIndex, uint8_t bBlockNumber, uint8_t bKeyType)

Authenticates a MIFARE Classic card with key and settings from previous PCSC Standard Extension commands.

Returns

Status code

Return values

<i>#PH_ERR_SUCCESS</i>	Operation successful.
------------------------	-----------------------

Parameters

in	<i>bCLSlotIndex</i>	Slot index
in	<i>bBlockNumber</i>	Block to authenticate
in	<i>bKeyType</i>	Use Key A or Key B

Definition at line 43 of file p3fw_pcsc.c.

4.6.1.4 `phStatus_t p3fw_pcsc_std_ext_prepare_l3_card (uint8_t bCLSlotIndex)`

Halts and reactivates a L3 card so that it is always in correct state to receive PCSC Standard Extension commands.

Returns

Status code

Return values

<code>#PH_ERR_SUCCESS</code>	Operation successful.
------------------------------	-----------------------

Parameters

<code>in</code>	<code>bCLSlotIndex</code>	Slot index
-----------------	---------------------------	------------

Definition at line 72 of file `p3fw_pcsc.c`.

Chapter 5

Data Structure Documentation

5.1 `_p3fw_cc_slot` Struct Reference

```
#include <p3fw_fw.h>
```

Data Fields

- enum [p3fw_slot_types](#) `eSlotType`
- enum [p3fw_pcsc_protocol](#) `eProtocolType`
- `uint8_t` [bPPP](#) [`P3FW_PCSC_PROTO_PARAMS_LEN_T1`]

5.1.1 Detailed Description

Definition at line 244 of file `p3fw_fw.h`.

5.1.2 Field Documentation

5.1.2.1 `uint8_t _p3fw_cc_slot::bPPP[P3FW_PCSC_PROTO_PARAMS_LEN_T1]`

PCSC Protocol Parameters

Definition at line 249 of file `p3fw_fw.h`.

5.1.2.2 `enum p3fw_pcsc_protocol _p3fw_cc_slot::eProtocolType`

Protocol type of PCSC

Definition at line 247 of file `p3fw_fw.h`.

5.1.2.3 enum p3fw_slot_types_p3fw_cc_slot::eSlotType

Type of card in slot

Definition at line 246 of file p3fw_fw.h.

The documentation for this struct was generated from the following file:

- [include/p3fw_fw.h](#)

5.2 _p3fw_cl_slot Struct Reference

Stores state of a contact less slot.

```
#include <p3fw_fw.h>
```

Data Fields

- enum [p3fw_slot_types](#) eSlotType
- enum [p3fw_pcsc_protocol](#) eProtocolType
- uint8_t [bBlockPoll](#)
- uint8_t [bSak](#)
- uint8_t [bUid](#) [P3FW_MAX_UID_A_LEN]
- uint8_t [bUidLength](#)
- uint8_t [bAts](#) [P3FW_MAX_ATS_LEN]
- uint8_t [bPPP](#) [P3FW_PCSC_PROTO_PARAMS_LEN_T1]
- void * [pHal](#)
- phpall14443p3a_Sw_DataParams_t [sISO14443L3a](#)
- phpall14443p3b_Sw_DataParams_t [sISO14443L3b](#)
- phpall14443p4a_Sw_DataParams_t [sISO14443L4a](#)
- phpall14443p4_Sw_DataParams_t [sISO14443L4](#)
- phpalMifare_Sw_DataParams_t [sMifare](#)
- phalMfc_Sw_DataParams_t [sMfc](#)
- phalMful_Sw_DataParams_t [sMful](#)

5.2.1 Detailed Description

Stores state of a contact less slot.

Definition at line 208 of file p3fw_fw.h.

5.2.2 Field Documentation

5.2.2.1 uint8_t _p3fw_cl_slot::bAts[P3FW_MAX_ATS_LEN]

Cards ATS

Definition at line 220 of file p3fw_fw.h.

5.2.2.2 uint8_t _p3fw_cl_slot::bBlockPoll

if set we do not do poll for this card

Definition at line 213 of file p3fw_fw.h.

5.2.2.3 uint8_t _p3fw_cl_slot::bPPP[P3FW_PCSC_PROTO_PARAMS_LEN_T1]

PCSC Protocol Parameters

Definition at line 221 of file p3fw_fw.h.

5.2.2.4 uint8_t _p3fw_cl_slot::bSak

Cards SAK

Definition at line 215 of file p3fw_fw.h.

5.2.2.5 uint8_t _p3fw_cl_slot::bUid[P3FW_MAX_UID_A_LEN]

UID (or PUPI) of the card

Definition at line 217 of file p3fw_fw.h.

5.2.2.6 uint8_t _p3fw_cl_slot::bUidLength

UID Length

Definition at line 218 of file p3fw_fw.h.

5.2.2.7 enum p3fw_pcsc_protocol _p3fw_cl_slot::eProtocolType

Protocol type of PCSC

Definition at line 211 of file p3fw_fw.h.

5.2.2.8 enum p3fw_slot_types _p3fw_cl_slot::eSlotType

Type of card in slot

Definition at line 210 of file p3fw_fw.h.

5.2.2.9 void* _p3fw_cl_slot::pHal

Generic HAL pointer (for use with higher layers)

Definition at line 231 of file p3fw_fw.h.

5.2.2.10 `phpall14443p3a_Sw_DataParams_t_p3fw_cl_slot::sISO14443L3a`

ISO14443-3A layer structure for current slot

Definition at line 233 of file `p3fw_fw.h`.

5.2.2.11 `phpall14443p3b_Sw_DataParams_t_p3fw_cl_slot::sISO14443L3b`

ISO14443-3B layer structure for current slot

Definition at line 234 of file `p3fw_fw.h`.

5.2.2.12 `phpall14443p4_Sw_DataParams_t_p3fw_cl_slot::sISO14443L4`

ISO14443-4 layer structure for current slot

Definition at line 236 of file `p3fw_fw.h`.

5.2.2.13 `phpall14443p4a_Sw_DataParams_t_p3fw_cl_slot::sISO14443L4a`

ISO14443-4A layer structure for current slot

Definition at line 235 of file `p3fw_fw.h`.

5.2.2.14 `phalMfc_Sw_DataParams_t_p3fw_cl_slot::sMfc`

MIFARE Classic PAL layer structure for current slot

Definition at line 239 of file `p3fw_fw.h`.

5.2.2.15 `phalMful_Sw_DataParams_t_p3fw_cl_slot::sMful`

MIFARE Ultralight PAL layer structure for current slot

Definition at line 240 of file `p3fw_fw.h`.

5.2.2.16 `phpalMifare_Sw_DataParams_t_p3fw_cl_slot::sMifare`

MIFARE PAL layer structure for current slot

Definition at line 238 of file `p3fw_fw.h`.

The documentation for this struct was generated from the following file:

- [include/p3fw_fw.h](#)

5.3 cpot_atr_frame Struct Reference

Data Fields

- uint8_t [state](#)
- uint8_t [historicalC](#)
- uint8_t [TAi](#)
- uint8_t [TBi](#)
- uint8_t [TCi](#)
- uint8_t [TDi](#)
- uint8_t [TCK](#)

5.3.1 Detailed Description

Definition at line 787 of file p3fw_samt1.c.

5.3.2 Field Documentation

5.3.2.1 uint8_t cpot_atr_frame::historicalC

Definition at line 790 of file p3fw_samt1.c.

5.3.2.2 uint8_t cpot_atr_frame::state

Definition at line 789 of file p3fw_samt1.c.

5.3.2.3 uint8_t cpot_atr_frame::TAi

Definition at line 791 of file p3fw_samt1.c.

5.3.2.4 uint8_t cpot_atr_frame::TBi

Definition at line 792 of file p3fw_samt1.c.

5.3.2.5 uint8_t cpot_atr_frame::TCi

Definition at line 793 of file p3fw_samt1.c.

5.3.2.6 uint8_t cpot_atr_frame::TCK

Definition at line 795 of file p3fw_samt1.c.

5.3.2.7 uint8_t cpot_atr_frame::TDi

Definition at line 794 of file p3fw_samt1.c.

The documentation for this struct was generated from the following file:

- src/p3fw_samt1.c

5.4 p2_fw_SAM_ctrl_Struct Reference

Data Fields

- uint8_t [mode](#)
- uint8_t [data](#) [P2_FW_SAM_MAX_REC_DATALEN]
- uint32_t [dataIndex](#)
- uint8_t [bitIndex](#)
- uint32_t [sendLen](#)
- uint8_t [chipType](#)
- uint8_t [chipMode](#)
- uint32_t [t0](#)
- uint32_t [etu](#)
- uint8_t [SessionATR](#) [P2_FW_SAM_MAX_ATR_SIZE]
- uint8_t [SessionATR_Size](#)
- uint8_t [conversion](#)
- uint8_t [sendSeqData](#)
- struct {
 - uint16_t [data](#) [500]
 - uint32_t [len](#)
 - uint32_t [dataIndex](#)
 - uint8_t [bitIndex](#)
 - uint8_t [parityCount](#)
 } [send_data](#)
- uint8_t [recExtraGuardTime](#)
- uint32_t [bwi](#)
- uint32_t [tmpByteWait_time](#)
- uint8_t [timingMode](#)
- uint32_t [cont_tim](#)

5.4.1 Detailed Description

Definition at line 98 of file p3fw_samt1.c.

5.4.2 Field Documentation

5.4.2.1 uint8_t p2_fw_SAM_ctrl::bitIndex

Definition at line 104 of file p3fw_samt1.c.

5.4.2.2 uint32_t p2_fw_SAM_ctrl::bwi

Definition at line 128 of file p3fw_samt1.c.

5.4.2.3 uint8_t p2_fw_SAM_ctrl::chipMode

Definition at line 108 of file p3fw_samt1.c.

5.4.2.4 uint8_t p2_fw_SAM_ctrl::chipType

Definition at line 107 of file p3fw_samt1.c.

5.4.2.5 uint32_t p2_fw_SAM_ctrl::cont_tim

Definition at line 132 of file p3fw_samt1.c.

5.4.2.6 uint8_t p2_fw_SAM_ctrl::conversion

Definition at line 115 of file p3fw_samt1.c.

5.4.2.7 uint8_t p2_fw_SAM_ctrl::data[P2_FW_SAM_MAX_REC_DATALEN]

Definition at line 102 of file p3fw_samt1.c.

5.4.2.8 uint16_t p2_fw_SAM_ctrl::data[500]

Definition at line 120 of file p3fw_samt1.c.

5.4.2.9 uint32_t p2_fw_SAM_ctrl::dataIndex

Definition at line 103 of file p3fw_samt1.c.

5.4.2.10 uint32_t p2_fw_SAM_ctrl::etu

Definition at line 110 of file p3fw_samt1.c.

5.4.2.11 uint32_t p2_fw_SAM_ctrl_::len

Definition at line 121 of file p3fw_samt1.c.

5.4.2.12 uint8_t p2_fw_SAM_ctrl_::mode

Definition at line 100 of file p3fw_samt1.c.

5.4.2.13 uint8_t p2_fw_SAM_ctrl_::parityCount

Definition at line 124 of file p3fw_samt1.c.

5.4.2.14 uint8_t p2_fw_SAM_ctrl_::recExtraGuardTime

Definition at line 127 of file p3fw_samt1.c.

5.4.2.15 struct { ... } p2_fw_SAM_ctrl_::send_data**5.4.2.16 uint32_t p2_fw_SAM_ctrl_::sendLen**

Definition at line 105 of file p3fw_samt1.c.

5.4.2.17 uint8_t p2_fw_SAM_ctrl_::sendSeqData

Definition at line 116 of file p3fw_samt1.c.

5.4.2.18 uint8_t p2_fw_SAM_ctrl_::SessionATR[P2_FW_SAM_MAX_ATR_SIZE]

Definition at line 112 of file p3fw_samt1.c.

5.4.2.19 uint8_t p2_fw_SAM_ctrl_::SessionATR_Size

Definition at line 113 of file p3fw_samt1.c.

5.4.2.20 uint32_t p2_fw_SAM_ctrl_::t0

Definition at line 109 of file p3fw_samt1.c.

5.4.2.21 uint8_t p2_fw_SAM_ctrl_::timingMode

Definition at line 131 of file p3fw_samt1.c.

5.4.2.22 uint32_t p2_fw_SAM_ctrl::tmpByteWait_time

Definition at line 129 of file p3fw_samt1.c.

The documentation for this struct was generated from the following file:

- [src/p3fw_samt1.c](#)

5.5 p3_fw_ccid_exec Struct Reference

Job parameters for.

```
#include <p3fw_fw.h>
```

Data Fields

- uint8_t [bSlot](#)
- void(* [bh](#))(uint8_t bCCIDSlotIndex)

5.5.1 Detailed Description

Job parameters for.

Definition at line 361 of file p3fw_fw.h.

5.5.2 Field Documentation

5.5.2.1 void(* p3_fw_ccid_exec::bh)(uint8_t bCCIDSlotIndex)

Function to call as bottom half handler

Definition at line 365 of file p3fw_fw.h.

5.5.2.2 uint8_t p3_fw_ccid_exec::bSlot

Slot number to use in bottom half handler

Definition at line 363 of file p3fw_fw.h.

The documentation for this struct was generated from the following file:

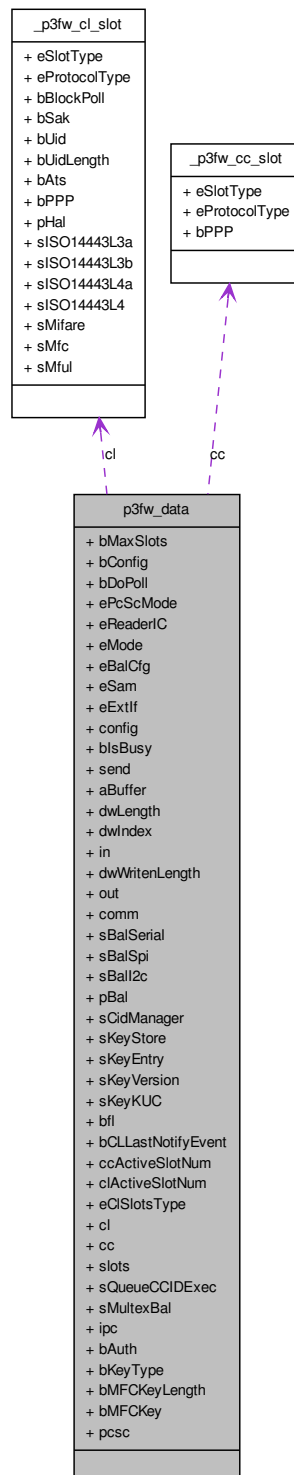
- [include/p3fw_fw.h](#)

5.6 p3fw_data Struct Reference

Stores all reader global settings and variables.

```
#include <p3fw_fw.h>
```

Collaboration diagram for p3fw_data:



Data Fields

- struct {
 - uint8_t bMaxSlots
 - uint8_t bConfig
 - uint8_t bDoPoll
 - enum p3fw_pcsc_mode ePcScMode
 - enum p3fw_ic eReaderIC
 - enum p3fw_mode eMode
 - enum p3fw_bal_configuration eBalCfg
 - enum p3fw_sam eSam
 - enum p3fw_external_interface eExtIf
 } config
- struct {
 - volatile uint8_t bIsBusy
 - uint32_t(* send)(uint8_t bWhere, uint8_t *pbWhat, uint32_t dwHowMuch)
 - struct {
 - uint8_t aBuffer [P3FW_MAX_CCID_BUFFER_LEN]
 - volatile uint32_t dwLength
 - volatile uint32_t dwIndex
 } in
 - struct {
 - uint8_t aBuffer [P3FW_MAX_CCID_BUFFER_LEN]
 - volatile uint32_t dwLength
 - volatile uint32_t dwWrittenLength
 } out
 } comm
- struct {
 - phbalReg_Lpc1768Serial_DataParams_t sBalSerial
 - phbalReg_Lpc1768Spi_DataParams_t sBalSpi
 - phbalReg_Lpc1768I2c_DataParams_t sBalI2c
 - void * pBal
 - phCidManager_Sw_DataParams_t sCidManager
 - phKeyStore_Sw_DataParams_t sKeyStore
 - phKeyStore_Sw_KeyEntry_t sKeyEntry [P3FW_KEYSTORE_NUM_KEYS]
 - phKeyStore_Sw_KeyVersionPair_t sKeyVersion [P3FW_KEYSTORE_NUM_KEYS *P3FW_KEYST
 - phKeyStore_Sw_KUCEntry_t sKeyKUC [P3FW_KEYSTORE_NUM_KEYS]
 } bfl
- struct {
 - uint8_t bCLLastNotifyEvent [P3FW_MAX_CONTACTLESS_SLOTS]
 - uint8_t ccActiveSlotNum
 - uint8_t clActiveSlotNum
 - enum p3fw_slot_types eCISlotsType
 - p3fw_cl_slot cl [P3FW_MAX_CONTACTLESS_SLOTS]
 - p3fw_cc_slot cc [P3FW_MAX_CONTACT_SLOTS]
 } slots

- struct {
 xQueueHandle sQueueCCIDExec
 xSemaphoreHandle sMultexBal
} ipc
- struct {
 uint8_t bAuth
 uint8_t bKeyType
 uint8_t bMFCKeyLength
 uint8_t bMFCKey [P3FW_MAX_KEY_LEN]
} pcsc

5.6.1 Detailed Description

Stores all reader global settings and variables.

Definition at line 255 of file p3fw_fw.h.

5.6.2 Field Documentation

5.6.2.1 uint8_t p3fw_data::aBuffer[P3FW_MAX_CCID_BUFFER_LEN]

Definition at line 291 of file p3fw_fw.h.

5.6.2.2 uint8_t p3fw_data::bAuth

Definition at line 350 of file p3fw_fw.h.

5.6.2.3 uint8_t p3fw_data::bCLLastNotifyEvent[P3FW_MAX_CONTACTLESS_SLOTS]

contains previous notification

Definition at line 331 of file p3fw_fw.h.

5.6.2.4 uint8_t p3fw_data::bConfig

historic field - do not use

Definition at line 263 of file p3fw_fw.h.

5.6.2.5 uint8_t p3fw_data::bDoPoll

Enable or disable polling for cards

Definition at line 264 of file p3fw_fw.h.

5.6.2.6 struct { ... } p3fw_data::bfl**5.6.2.7 volatile uint8_t p3fw_data::bIsBusy**

flag for setting CCID state machine to busy

Definition at line 281 of file p3fw_fw.h.

5.6.2.8 uint8_t p3fw_data::bKeyType

Definition at line 351 of file p3fw_fw.h.

5.6.2.9 uint8_t p3fw_data::bMaxSlots

number of slots to be reported

Definition at line 262 of file p3fw_fw.h.

5.6.2.10 uint8_t p3fw_data::bMFCKey[P3FW_MAX_KEY_LEN]

Definition at line 353 of file p3fw_fw.h.

5.6.2.11 uint8_t p3fw_data::bMFCKeyLength

Definition at line 352 of file p3fw_fw.h.

5.6.2.12 p3fw_cc_slot p3fw_data::cc[P3FW_MAX_CONTACT_SLOTS]

contains contact slots

Definition at line 339 of file p3fw_fw.h.

5.6.2.13 uint8_t p3fw_data::ccActiveSlotNum

Definition at line 333 of file p3fw_fw.h.

5.6.2.14 p3fw_cl_slot p3fw_data::cl[P3FW_MAX_CONTACTLESS_SLOTS]

contains contac-less slots

Definition at line 337 of file p3fw_fw.h.

5.6.2.15 uint8_t p3fw_data::clActiveSlotNum

number of active contact less slots

Definition at line 334 of file p3fw_fw.h.

5.6.2.16 struct { ... } p3fw_data::comm

5.6.2.17 struct { ... } p3fw_data::config

5.6.2.18 volatile uint32_t p3fw_data::dwIndex

Definition at line 293 of file p3fw_fw.h.

5.6.2.19 volatile uint32_t p3fw_data::dwLength

Definition at line 292 of file p3fw_fw.h.

5.6.2.20 volatile uint32_t p3fw_data::dwWrittenLength

Definition at line 303 of file p3fw_fw.h.

5.6.2.21 enum p3fw_bal_configuration p3fw_data::eBalCfg

after config read contains internal interface

Definition at line 271 of file p3fw_fw.h.

5.6.2.22 enum p3fw_slot_types p3fw_data::eCISlotsType

card type in slots

Definition at line 336 of file p3fw_fw.h.

5.6.2.23 enum p3fw_external_interface p3fw_data::eExtIf

after config read contains external interface

Definition at line 273 of file p3fw_fw.h.

5.6.2.24 enum p3fw_mode p3fw_data::eMode

after config read contains reader mode

Definition at line 270 of file p3fw_fw.h.

5.6.2.25 enum p3fw_pcsc_mode p3fw_data::ePcScMode

type of PCSC mode we are in

Definition at line 266 of file p3fw_fw.h.

5.6.2.26 enum p3fw_ic p3fw_data::eReaderIC

after config read contains reader IC

Definition at line 268 of file p3fw_fw.h.

5.6.2.27 enum p3fw_sam p3fw_data::eSam

after config read contains how to use SAM

Definition at line 272 of file p3fw_fw.h.

5.6.2.28 struct { ... } p3fw_data::in**5.6.2.29 struct { ... } p3fw_data::ipc****5.6.2.30 struct { ... } p3fw_data::out****5.6.2.31 void* p3fw_data::pBal**

generic BAL interface to be used by HAL

Definition at line 314 of file p3fw_fw.h.

5.6.2.32 struct { ... } p3fw_data::pcsc**5.6.2.33 phbalReg_Lpc1768I2c_DataParams_t p3fw_data::sBalI2c**

BAL structure for LCP1768 I2C interface

Definition at line 312 of file p3fw_fw.h.

5.6.2.34 phbalReg_Lpc1768Serial_DataParams_t p3fw_data::sBalSerial

BAL structure for LCP1768 Serial interface

Definition at line 310 of file p3fw_fw.h.

5.6.2.35 phbalReg_Lpc1768Spi_DataParams_t p3fw_data::sBalSpi

BAL structure for LCP1768 SPI interface

Definition at line 311 of file p3fw_fw.h.

5.6.2.36 phCidManager_Sw_DataParams_t p3fw_data::sCidManager

CID Manager structure

Definition at line 320 of file p3fw_fw.h.

5.6.2.37 uint32_t(* p3fw_data::send)(uint8_t bWhere, uint8_t *pbWhat, uint32_t dwHowMuch)

sending callback - must be implemented by external communication interface

Definition at line 283 of file p3fw_fw.h.

5.6.2.38 phKeyStore_Sw_KeyEntry_t p3fw_data::sKeyEntry[P3FW_KEYSTORE_NUM_KEYS]

contains all key entries

Definition at line 323 of file p3fw_fw.h.

5.6.2.39 phKeyStore_Sw_KUCEntry_t p3fw_data::sKeyKUC[P3FW_KEYSTORE_NUM_KEYS]

contains all kucs

Definition at line 325 of file p3fw_fw.h.

5.6.2.40 phKeyStore_Sw_DataParams_t p3fw_data::sKeyStore

main key store structure

Definition at line 322 of file p3fw_fw.h.

5.6.2.41 phKeyStore_Sw_KeyVersionPair_t p3fw_data::sKeyVersion[P3FW_KEYSTORE_NUM_KEYS * P3FW_KEYSTORE_NUM_VERS]

contains all key versions

Definition at line 324 of file p3fw_fw.h.

5.6.2.42 struct { ... } p3fw_data::slots**5.6.2.43 xSemaphoreHandle p3fw_data::sMultexBal**

BAL lock

Definition at line 345 of file p3fw_fw.h.

5.6.2.44 xQueueHandle p3fw_data::sQueueCCIDExec

queue for CCID execution task

Definition at line 344 of file p3fw_fw.h.

The documentation for this struct was generated from the following file:

- [include/p3fw_fw.h](#)

5.7 p3fw_hal_buffer Struct Reference

Every HAL requires buffer for sending and receiving.

```
#include <p3fw_fw.h>
```

Data Fields

- uint8_t [tx](#) [P3FW_MAX_HAL_TX_BUFFER]
- uint8_t [rx](#) [P3FW_MAX_HAL_RX_BUFFER]

5.7.1 Detailed Description

Every HAL requires buffer for sending and receiving.

Definition at line 199 of file p3fw_fw.h.

5.7.2 Field Documentation

5.7.2.1 uint8_t p3fw_hal_buffer::rx[P3FW_MAX_HAL_RX_BUFFER]

Definition at line 202 of file p3fw_fw.h.

5.7.2.2 uint8_t p3fw_hal_buffer::tx[P3FW_MAX_HAL_TX_BUFFER]

Definition at line 201 of file p3fw_fw.h.

The documentation for this struct was generated from the following file:

- [include/p3fw_fw.h](#)

5.8 sam_t1_param Struct Reference

```
#include <p3fw_samt1.h>
```

Data Fields

- uint8_t [FI_DI](#)
- uint8_t [GuardTime](#)
- uint8_t [BWI_CWI](#)
- uint8_t [ClockStop](#)
- uint8_t [IFSC](#)

5.8.1 Detailed Description

Definition at line 34 of file p3fw_samt1.h.

5.8.2 Field Documentation

5.8.2.1 uint8_t sam_t1_param::BWI_CWI

Definition at line 38 of file p3fw_samt1.h.

5.8.2.2 uint8_t sam_t1_param::ClockStop

Definition at line 39 of file p3fw_samt1.h.

5.8.2.3 uint8_t sam_t1_param::FI_DI

Definition at line 36 of file p3fw_samt1.h.

5.8.2.4 uint8_t sam_t1_param::GuardTime

Definition at line 37 of file p3fw_samt1.h.

5.8.2.5 uint8_t sam_t1_param::IFSC

Definition at line 40 of file p3fw_samt1.h.

The documentation for this struct was generated from the following file:

- include/[p3fw_samt1.h](#)

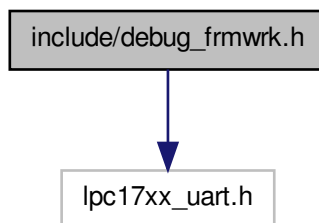
Chapter 6

File Documentation

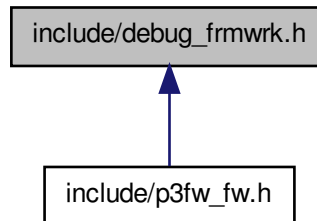
6.1 include/debug_frmwrk.h File Reference

```
#include "lpc17xx_uart.h"
```

Include dependency graph for debug_frmwrk.h:



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



Defines

- #define [DEBUG_UART_PORT](#) LPC_UART0
- #define [_DBGH_](#)(x, y)
- #define [_DBG](#)(x)
- #define [_DBG_](#)(x)
- #define [_DBC](#)(x)
- #define [_DBD](#)(x)
- #define [_DBD16](#)(x)
- #define [_DBD32](#)(x)
- #define [_DBH](#)(x)
- #define [_DBH16](#)(x)
- #define [_DBH32](#)(x)
- #define [_DG](#)
- #define [_DBGL](#)(x) [_db_msg](#) ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x)
- #define [_DBGL_](#)(x) [_db_msg_](#) ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x)
- #define [_DBGHL_](#)(x, y) [_db_mhex](#) ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x, y)
- #define [_DBH16L](#)(x) [_db_hex_16](#) ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x)
- #define [_DBH32L](#)(x) [_db_hex_32](#) ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x)

Functions

- void [UARTPutHexMulti](#) (LPC_UART_TypeDef *UARTx, uint8_t *s, uint32_t len)

- void [UARTPutChar](#) (LPC_UART_TypeDef *UARTx, uint8_t ch)
Puts a character to UART port.
- void [UARTPuts](#) (LPC_UART_TypeDef *UARTx, const void *str)
Puts a string to UART port.
- void [UARTPuts_](#) (LPC_UART_TypeDef *UARTx, const void *str)
Puts a string to UART port and print new line.
- void [UARTPutDec](#) (LPC_UART_TypeDef *UARTx, uint8_t decnum)
Puts a decimal number to UART port.
- void [UARTPutDec16](#) (LPC_UART_TypeDef *UARTx, uint16_t decnum)
Puts a decimal number to UART port.
- void [UARTPutDec32](#) (LPC_UART_TypeDef *UARTx, uint32_t decnum)
Puts a decimal number to UART port.
- void [UARTPutHex](#) (LPC_UART_TypeDef *UARTx, uint8_t hexnum)
Puts a hex number to UART port.
- void [UARTPutHex16](#) (LPC_UART_TypeDef *UARTx, uint16_t hexnum)
Puts a hex number to UART port.
- void [UARTPutHex32](#) (LPC_UART_TypeDef *UARTx, uint32_t hexnum)
Puts a hex number to UART port.
- uint8_t [UARTGetChar](#) (LPC_UART_TypeDef *UARTx)
Get a character to UART port.
- void [debug_frmwrk_init](#) (void)
Initialize Debug frame work through initializing UART port.

Variables

- void(* [_db_mhex](#))(LPC_UART_TypeDef *UARTx, uint8_t *s, uint32_t len)
- void(* [_db_msg](#))(LPC_UART_TypeDef *UARTx, const void *s)
- void(* [_db_msg_](#))(LPC_UART_TypeDef *UARTx, const void *s)
- void(* [_db_char](#))(LPC_UART_TypeDef *UARTx, uint8_t ch)
- void(* [_db_dec](#))(LPC_UART_TypeDef *UARTx, uint8_t decn)
- void(* [_db_dec_16](#))(LPC_UART_TypeDef *UARTx, uint16_t decn)
- void(* [_db_dec_32](#))(LPC_UART_TypeDef *UARTx, uint32_t decn)
- void(* [_db_hex](#))(LPC_UART_TypeDef *UARTx, uint8_t hexn)
- void(* [_db_hex_16](#))(LPC_UART_TypeDef *UARTx, uint16_t hexn)
- void(* [_db_hex_32](#))(LPC_UART_TypeDef *UARTx, uint32_t hexn)
- uint8_t(* [_db_get_char](#))(LPC_UART_TypeDef *UARTx)

6.1.1 Define Documentation

6.1.1.1 `#define _DBC(x)`

Definition at line 62 of file `debug_frmwrk.h`.

6.1.1.2 `#define _DBD(x)`

Definition at line 63 of file `debug_frmwrk.h`.

6.1.1.3 `#define _DBD16(x)`

Definition at line 64 of file `debug_frmwrk.h`.

6.1.1.4 `#define _DBD32(x)`

Definition at line 65 of file `debug_frmwrk.h`.

6.1.1.5 `#define _DBG(x)`

Definition at line 60 of file `debug_frmwrk.h`.

6.1.1.6 `#define _DBG_(x)`

Definition at line 61 of file `debug_frmwrk.h`.

6.1.1.7 `#define _DBGH_(x, y)`

Definition at line 59 of file `debug_frmwrk.h`.

6.1.1.8 `#define _DBGHL_(x, y) _db_mhex ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x, y)`

Definition at line 72 of file `debug_frmwrk.h`.

6.1.1.9 `#define _DBGL(x) _db_msg ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x)`

Definition at line 70 of file `debug_frmwrk.h`.

6.1.1.10 `#define _DBGL_(x) _db_msg_ ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x)`

Definition at line 71 of file `debug_frmwrk.h`.

6.1.1.11 #define _DBH(x)

Definition at line 66 of file debug_frmwrk.h.

6.1.1.12 #define _DBH16(x)

Definition at line 67 of file debug_frmwrk.h.

6.1.1.13 #define _DBH16L(x) _db_hex_16 ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x)

Definition at line 73 of file debug_frmwrk.h.

6.1.1.14 #define _DBH32(x)

Definition at line 68 of file debug_frmwrk.h.

6.1.1.15 #define _DBH32L(x) _db_hex_32 ((LPC_UART_TypeDef *)DEBUG_UART_PORT, x)

Definition at line 74 of file debug_frmwrk.h.

6.1.1.16 #define _DG

Definition at line 69 of file debug_frmwrk.h.

6.1.1.17 #define DEBUG_UART_PORT LPC_UART0

Definition at line 34 of file debug_frmwrk.h.

6.1.2 Function Documentation**6.1.2.1 void debug_frmwrk_init (void)**

Initialize Debug frame work through initializing UART port.

Parameters

in	None	
----	------	--

Returns

None

Definition at line 242 of file debug_frmwrk.c.

6.1.2.2 uint8_t UARTGetChar (LPC_UART_TypeDef * UARTx)

Get a character to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>ch</i>	Character to put

Returns

None

Definition at line 77 of file debug_frmwrk.c.

6.1.2.3 void UARTPutChar (LPC_UART_TypeDef * UARTx, uint8_t ch)

Puts a character to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>ch</i>	Character to put

Returns

None

Definition at line 65 of file debug_frmwrk.c.

6.1.2.4 void UARTPutDec (LPC_UART_TypeDef * UARTx, uint8_t decnum)

Puts a decimal number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>decnum</i>	Decimal number (8-bit long)

Returns

None

Definition at line 121 of file debug_frmwrk.c.

6.1.2.5 void UARTPutDec16 (LPC_UART_TypeDef * UARTx, uint16_t decnum)

Puts a decimal number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>decnum</i>	Decimal number (8-bit long)

Returns

None

Definition at line 137 of file debug_frmwrk.c.

6.1.2.6 void UARTPutDec32 (LPC_UART_TypeDef * *UARTx*, uint32_t *decnum*)

Puts a decimal number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>decnum</i>	Decimal number (8-bit long)

Returns

None

Definition at line 157 of file debug_frmwrk.c.

6.1.2.7 void UARTPutHex (LPC_UART_TypeDef * *UARTx*, uint8_t *hexnum*)

Puts a hex number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>hexnum</i>	Hex number (8-bit long)

Returns

None

Definition at line 187 of file debug_frmwrk.c.

6.1.2.8 void UARTPutHex16 (LPC_UART_TypeDef * *UARTx*, uint16_t *hexnum*)

Puts a hex number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>hexnum</i>	Hex number (16-bit long)

Returns

None

Definition at line 206 of file debug_frmwrk.c.

6.1.2.9 void UARTPutHex32 (LPC_UART_TypeDef * *UARTx*, uint32_t *hexnum*)

Puts a hex number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>hexnum</i>	Hex number (32-bit long)

Returns

None

Definition at line 224 of file debug_frmwrk.c.

6.1.2.10 void UARTPutHexMulti (LPC_UART_TypeDef * *UARTx*, uint8_t * *s*, uint32_t *len*)

Definition at line 40 of file debug_frmwrk.c.

6.1.2.11 void UARTPuts (LPC_UART_TypeDef * *UARTx*, const void * *str*)

Puts a string to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>str</i>	string to put

Returns

None

Definition at line 91 of file debug_frmwrk.c.

6.1.2.12 void UARTPuts_ (LPC_UART_TypeDef * *UARTx*, const void * *str*)

Puts a string to UART port and print new line.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>str</i>	String to put

Returns

None

Definition at line 108 of file debug_frmwrk.c.

6.1.3 Variable Documentation**6.1.3.1 void(*_db_char)(LPC_UART_TypeDef *UARTx, uint8_t ch)**

Definition at line 31 of file debug_frmwrk.c.

6.1.3.2 void(*_db_dec)(LPC_UART_TypeDef *UARTx, uint8_t decn)

Definition at line 32 of file debug_frmwrk.c.

6.1.3.3 void(*_db_dec_16)(LPC_UART_TypeDef *UARTx, uint16_t decn)

Definition at line 33 of file debug_frmwrk.c.

6.1.3.4 void(*_db_dec_32)(LPC_UART_TypeDef *UARTx, uint32_t decn)

Definition at line 34 of file debug_frmwrk.c.

6.1.3.5 uint8_t(*_db_get_char)(LPC_UART_TypeDef *UARTx)

Definition at line 38 of file debug_frmwrk.c.

6.1.3.6 void(*_db_hex)(LPC_UART_TypeDef *UARTx, uint8_t hexn)

Definition at line 35 of file debug_frmwrk.c.

6.1.3.7 void(*_db_hex_16)(LPC_UART_TypeDef *UARTx, uint16_t hexn)

Definition at line 36 of file debug_frmwrk.c.

6.1.3.8 void(*_db_hex_32)(LPC_UART_TypeDef *UARTx, uint32_t hexn)

Definition at line 37 of file debug_frmwrk.c.

6.1.3.9 void(*_db_mhex)(LPC_UART_TypeDef *UARTx, uint8_t *s, uint32_t len)

Definition at line 28 of file debug_frmwrk.c.

6.1.3.10 void(*_db_msg)(LPC_UART_TypeDef *UARTx, const void *s)

Definition at line 29 of file debug_frmwrk.c.

6.1.3.11 void(*_db_msg_)(LPC_UART_TypeDef *UARTx, const void *s)

Definition at line 30 of file debug_frmwrk.c.

6.2 include/p3fw_apdu.h File Reference

Defines

- #define [P3FW_APDU_PAYLOAD](#) 0
- #define [P3FW_APDU_CC_EXT_MANAGE_SESSION](#) 0x00
- #define [P3FW_APDU_CC_EXT_TRANS_EXCHANGE](#) 0x01
- #define [P3FW_APDU_CC_EXT_SWITCH_PROTOCOL](#) 0x02

ISO7816 APDUs offsets

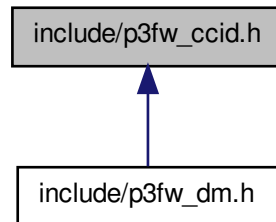
- #define [P3FW_APDU_CLASS](#) 0
- #define [P3FW_APDU_INS](#) 1
- #define [P3FW_APDU_P1](#) 2
- #define [P3FW_APDU_P2](#) 3
- #define [P3FW_APDU_Lc](#) 4
- #define [P3FW_APDU_Le](#) 4

PCSC Memory Cards Extensions

- #define [P3FW_APDU_CC_EXT_INS](#) 0xC2
- #define [P3FW_APDU_GET_DATA_INS](#) 0xCA
- #define [P3FW_APDU_LOAD_KEY_INS](#) 0x82
- #define [P3FW_APDU_G_AUTH_CMD_INS](#) 0x86
- #define [P3FW_APDU_READ_BIN](#) 0xB0
- #define [P3FW_APDU_UPDATE_BIN](#) 0xD6

6.3 include/p3fw_ccid.h File Reference

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



Defines

- #define P3FW_CCID_BULK_HEADER 0x0A
- #define P3FW_CCID_HEADER_MESSAGE_TYPE 0x00
- #define P3FW_CCID_HEADER_LENGTH_BYTE_1 0x01
- #define P3FW_CCID_HEADER_LENGTH_BYTE_2 0x02
- #define P3FW_CCID_HEADER_LENGTH_BYTE_3 0x03
- #define P3FW_CCID_HEADER_LENGTH_BYTE_4 0x04
- #define P3FW_CCID_HEADER_SLOT 0x05
- #define P3FW_CCID_HEADER_SEQ 0x06
- #define P3FW_CCID_HEADER_MSG_BYTE_1 0x07
- #define P3FW_CCID_HEADER_MSG_BYTE_2 0x08
- #define P3FW_CCID_HEADER_MSG_BYTE_3 0x09
- #define P3FW_CCID_RDR_TO_PC_SLOT_DATA_BLOCK 0x80
- #define P3FW_CCID_RDR_TO_PC_SLOT_STATUS 0x81
- #define P3FW_CCID_RDR_TO_PC_PARAMETERS 0x82
- #define P3FW_CCID_RDR_TO_PC_ESCAPE 0x83
- #define P3FW_CCID_STATUS_CMD_FAILED 0x40
- #define P3FW_CCID_ERROR_SLOT_BUSY 0xE0
- #define P3FW_CCID_ERROR_SLOT_NOT_EXIST 0x05
- #define P3FW_CCID_ERROR_SLOT_ICC_MUTE 0xFE
- #define P3FW_CCID_ERROR_SLOT_XFR_OVERRUN 0xFC
- #define P3FW_CCID_ERROR_SLOT_CMD_NOT_SUPPORTED 0x00
- #define P3FW_CCID_ERROR_SLOT_HW_ERROR 0xFB
- #define P3FW_CCID_BULK_OUT_REQ_ICCPowerON 0x62
- #define P3FW_CCID_BULK_OUT_REQ_ICCPowerOFF 0x63
- #define P3FW_CCID_BULK_OUT_REQ_GETSLOTSTATUS 0x65

- #define [P3FW_CCID_BULK_OUT_REQ_XFRBLOCK](#) 0x6F
- #define [P3FW_CCID_BULK_OUT_REQ_GETPARAMETERS](#) 0x6C
- #define [P3FW_CCID_BULK_OUT_REQ_SETPARAMETERS](#) 0x61
- #define [P3FW_CCID_BULK_OUT_REQ_ESCAPE](#) 0x6B
- #define [P3FW_CCID_INT_IN_NOTIFY_SLOT_CHANGE](#) 0x50
- #define [P3FW_CCID_CARD_IN_SLOT](#) 0x01
- #define [P3FW_CCID_CARD_OUT_SLOT](#) 0x00

Functions

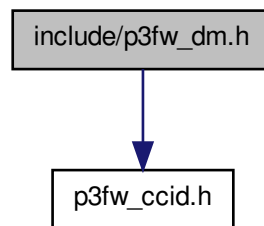
- void [p3fw_ccid_th_dispatch](#) (void)
Main function for top half CCID handler.
- void [p3fw_ccid_send_frame](#) (uint8_t bMessageType, uint8_t bByte1, uint8_t bByte2, uint8_t bByte3)
Sends CCID frame over USB to the driver.
- uint8_t * [p3fw_ccid_get_output_payload_buffer](#) ()
Returns a pointer to CCID internal output buffer.
- void [p3fw_ccid_set_output_payload_length](#) (uint32_t dwPayloadLength)
Sets the payload length of the output CCID message.
- uint8_t * [p3fw_ccid_get_input_payload_buffer](#) ()
Returns a pointer to CCID internal input buffer.
- uint32_t [p3fw_ccid_get_input_payload_length](#) ()
Gets the payload length of the input CCID message.
- uint8_t [p3fw_ccid_input_header_get_byte](#) (uint8_t bByte)
Gets one byte from input header.
- void [p3fw_ccid_bh_get_slot_status](#) (uint8_t bCCIDSlotIndex)
Returns /c bCCIDSlotIndex status.
- void [p3fw_ccid_bh_icc_power_on](#) (uint8_t bCCIDSlotIndex)
Card Power On.
- void [p3fw_ccid_bh_icc_power_off](#) (uint8_t bCCIDSlotIndex)
Card Power Off.
- void [p3fw_ccid_bh_get_parameters](#) (uint8_t bCCIDSlotIndex)
CCID Get Parameters.
- void [p3fw_ccid_bh_set_parameters](#) (uint8_t bCCIDSlotIndex)
CCID Set Parameters.

- void [p3fw_ccid_bh_escape](#) (uint8_t bCCIDSlotIndex)
CCID Escape Interface (SCardControl)
- void [p3fw_ccid_bh_xfer](#) (uint8_t bCCIDSlotIndex)
CCID Transfer command.
- void [p3fw_ccid_send_notify](#) ()
Sends slot status notification through INT end-point.

6.4 include/p3fw_dm.h File Reference

```
#include <p3fw_ccid.h>
```

Include dependency graph for p3fw_dm.h:



Defines

- #define [P3FW_DM_CID](#) 0x90
- #define [P3FW_DM_RO](#) 0xA0
- #define [P3FW_DM_HAL](#) 0xB0
- #define [P3FW_DM_L3](#) 0xC0
- #define [P3FW_DM_L4A](#) 0xD0
- #define [P3FW_DM_L4](#) 0xE0
- #define [P3FW_DM_XCHG](#) 0xF0
- #define [P3FW_DM_KSTOR](#) 0x70
- #define [P3FW_DM_CONTACT_CARD](#) 0x80
- #define [P3FW_DM_RO_LEDS_OFF](#) 0x01
- #define [P3FW_DM_RO_LEDS_ON](#) 0x02
- #define [P3FW_DM_RO_RESET](#) 0x03

- #define P3FW_DM_RO_CONF_OVER 0x04
- #define P3FW_DM_RO_SET_CONF 0x05
- #define P3FW_DM_RO_GET_CONF 0x0C
- #define P3FW_DM_RO_GET_STATUS 0x06
- #define P3FW_DM_RO_READ_REG 0x07
- #define P3FW_DM_RO_WRITE_REG 0x08
- #define P3FW_DM_RO_FIELD_ON 0x09
- #define P3FW_DM_RO_FIELD_OFF 0x0A
- #define P3FW_DM_RO_FIELD_RESET 0x0B
- #define P3FW_DM_RO_SET_PCSC_MODE 0x0D
- #define P3FW_DM_RO_TEST_MODE 0x0E
- #define P3FW_DM_HAL_INIT 0x01
- #define P3FW_DM_HAL_XCHG 0x02
- #define P3FW_DM_HAL_SET_CFG 0x03
- #define P3FW_DM_HAL_GET_CFG 0x04
- #define P3FW_DM_HAL_APP_PROT_SET 0x05
- #define P3FW_DM_HAL_WAIT 0x06
- #define P3FW_DM_HAL_MFC_AUTH 0x07
- #define P3FW_DM_HAL_EXEC_CMD 0x08
- #define P3FW_DM_HAL_MFC_AUTH_KEY 0x09
- #define P3FW_DM_L3_INIT 0x01
- #define P3FW_DM_L3_REQA 0x02
- #define P3FW_DM_L3_WKUA 0x03
- #define P3FW_DM_L3_HLTA 0x04
- #define P3FW_DM_L3_ANTICOL 0x05
- #define P3FW_DM_L3_SELECT 0x06
- #define P3FW_DM_L3_ACT_CARD 0x07
- #define P3FW_DM_L3_XCHG 0x08
- #define P3FW_DM_L3_GET_SER 0x09
- #define P3FW_DM_L4A_INIT 0x01
- #define P3FW_DM_L4A_RATS 0x02
- #define P3FW_DM_L4A_PPS 0x03
- #define P3FW_DM_L4A_ACT_CARD 0x04
- #define P3FW_DM_L4A_GET_PROTO_PARM 0x05
- #define P3FW_DM_L4_INIT 0x01
- #define P3FW_DM_L4_SET_PROTO 0x02
- #define P3FW_DM_L4_RESET_PROTO 0x03
- #define P3FW_DM_L4_DESELECT 0x04
- #define P3FW_DM_L4_PRES_CHECK 0x05
- #define P3FW_DM_L4_XCHG 0x06
- #define P3FW_DM_L4_SET_CFG 0x07
- #define P3FW_DM_L4_GET_CFG 0x08
- #define P3FW_DM_XCHG_L3 0x01
- #define P3FW_DM_XCHG_L4 0x02
- #define P3FW_DM_XCHG_PC 0x03
- #define P3FW_DM_XCHG_RAW 0x04

- #define [P3FW_DM_XCHG_MFC_AUTH](#) 0x05
- #define [P3FW_DM_XCHG_MFC_AUTH_KEY](#) 0x06
- #define [P3FW_DM_XCHG_INIT](#) 0x07
- #define [P3FW_DM_CID_GET_FREE](#) 0x01
- #define [P3FW_DM_CID_FREE](#) 0x02
- #define [P3FW_DM_CID_INIT](#) 0x03
- #define [P3FW_DM_KSTOR_INIT](#) 0x01
- #define [P3FW_DM_KSTOR_FORMAT_KEY](#) 0x02
- #define [P3FW_DM_KSTOR_SET_KEY](#) 0x03
- #define [P3FW_DM_KSTOR_SET_KEY_POS](#) 0x04
- #define [P3FW_DM_KSTOR_SET_KUC](#) 0x05
- #define [P3FW_DM_KSTOR_SET_FULL_KEY](#) 0x07
- #define [P3FW_DM_KSTOR_GET_KEY_ENTRY](#) 0x08
- #define [P3FW_DM_KSTOR_GET_KEY](#) 0x09
- #define [P3FW_DM_KSTOR_SET_CONFIG](#) 0x0A
- #define [P3FW_DM_KSTOR_GET_CONFIG](#) 0x0B
- #define [P3FW_DM_KSTOR_CHG_KUC](#) 0x0C
- #define [P3FW_DM_KSTOR_GET_KUC](#) 0x0D
- #define [P3FW_DM_KSTOR_SET_CFG_STR](#) 0x0E
- #define [P3FW_DM_KSTOR_GET_CFG_STR](#) 0x0F
- #define [P3FW_DM_CONTACTCARD_ACTIVATE_CARD](#) 0x01
- #define [P3FW_DM_CONTACTCARD_COLD_RESET](#) 0x02
- #define [P3FW_DM_CONTACTCARD_WARM_RESET](#) 0x03
- #define [P3FW_DM_CONTACTCARD_CLOCK_STOP](#) 0x04
- #define [P3FW_DM_CONTACTCARD_CLOCK_START](#) 0x05
- #define [P3FW_DM_CONTACTCARD_DEACTIVATE_CARD](#) 0x06
- #define [P3FW_DM_CONTACTCARD_PRESENCE_CHECK](#) 0x07
- #define [P3FW_DM_CONTACTCARD_TRANSMIT_DATA](#) 0x08
- #define [P3FW_DM_CONTACTCARD_PPS](#) 0x09
- #define [P3FW_DM_CHECK_STATUS](#)(status) if (PH_ERR_SUCCESS == status) { p3fw_dm_send_frame (status, 0); return; }
- #define [P3FW_DM_CHECK_LENGTH_EQUAL](#)(len) if ((len != p3fw_dm_get_input_payload_length()) || (p3fw_dm_get_input_payload_length() != (p3fw_ccid_get_input_payload_length() - 4))) { p3fw_dm_send_frame (PH_COMP_BAL | PH_ERR_LENGTH_ERROR, 0); return; }
- #define [P3FW_DM_CHECK_LENGTH_LARGER](#)(len) if ((len > p3fw_dm_get_input_payload_length()) || (p3fw_dm_get_input_payload_length() != (p3fw_ccid_get_input_payload_length() - 4))) { p3fw_dm_send_frame (PH_COMP_BAL | PH_ERR_LENGTH_ERROR, 0); return; }

Functions

- void [p3fw_dm_xfer](#) ()
Direct Mode Transfer Mode.
- void [p3fw_dm_ro](#) ()

Direct Mode Reader Operations (also available through SCardControl/CCID Escape Interface)

- void [p3fw_dm_hal](#) ()
Direct Mode Hardware Abstraction Layer Functions.
- void [p3fw_dm_l3](#) ()
Direct Mode ISO14443-3 Functions.
- void [p3fw_dm_l4a](#) ()
Direct Mode ISO14443-4A Functions.
- void [p3fw_dm_l4](#) ()
Direct Mode ISO14443-4 Functions.
- void [p3fw_dm_xchg](#) ()
Direct Mode Exchange Functions.
- void [p3fw_dm_cid](#) ()
Direct Mode CID Management Functions.
- void [p3fw_dm_keystore](#) ()
Direct Mode Keystore Management Functions.
- void [p3fw_dm_contact_card](#) ()
Direct Mode Contact Card Functions.
- uint8_t [p3fw_dm_get_class](#) ()
Extracts class from the frame.
- uint8_t [p3fw_dm_get_instruction](#) ()
Extracts instruction from the frame.
- uint8_t [p3fw_dm_get_slot_index](#) ()
Extracts slot index from the frame.
- uint16_t [p3fw_dm_get_input_payload_length](#) ()
Gets input length.
- uint8_t * [p3fw_dm_get_input_payload](#) ()
Gets input payload buffer.
- uint8_t * [p3fw_dm_get_output_payload](#) ()
Gets output payload buffer.
- uint8_t * [p3fw_dm_get_output_pointer_uint8_t](#) (uint16_t wDmByteIndex)

- uint8_t [p3fw_dm_get_input_uint8_t](#) (uint16_t wDmByteIndex)
- uint16_t [p3fw_dm_get_input_uint16_t](#) (uint16_t wDmByteIndex)
- uint32_t [p3fw_dm_get_input_uint32_t](#) (uint16_t wDmByteIndex)
- uint8_t * [p3fw_dm_get_input_pointer_uint8_t](#) (uint16_t wDmByteIndex)
- void [p3fw_dm_send_frame](#) (phStatus_t status, uint16_t wFrameLength)

Sends frame.

6.5 include/p3fw_fw.h File Reference

```
#include <string.h>
#include <FreeRTOS.h>
#include <task.h>
#include <queue.h>
#include <semphr.h>
#include <ph_Status.h>
#include <phbalReg.h>
#include <phhalHw.h>
#include <phCidManager.h>
#include <phpalI14443p3a.h>
#include <phpalI14443p3b.h>
#include <phpalI14443p4a.h>
#include <phpalI14443p4.h>
#include <phKeyStore.h>
#include <phpalMifare.h>
#include <phalMfc.h>
#include <phalMful.h>
#include <debug_frmwrk.h>
```

Include dependency graph for p3fw_fw.h:



Data Structures

- struct [p3fw_hal_buffer](#)
Every HAL requires buffer for sending and receiving.
- struct [_p3fw_cl_slot](#)
Stores state of a contact less slot.
- struct [_p3fw_cc_slot](#)
- struct [p3fw_data](#)
Stores all reader global settings and variables.
- struct [p3_fw_ccid_exec](#)
Job parameters for.

Defines

- #define [P3FW_MAX_CCID_BUFFER_LEN](#) 271
- #define [P3FW_MAX_CONTACTLESS_SLOTS](#) 14
- #define [P3FW_MAX_CONTACT_SLOTS](#) 1
- #define [P3FW_CONTACT_SLOTS_MASK](#) 0x80
- #define [P3FW_MAX_UID_A_LEN](#) 10
- #define [P3FW_MAX_UID_B_LEN](#) 4
- #define [P3FW_MAX_ATQB_LEN](#) 14
- #define [P3FW_MAX_HAL_TX_BUFFER](#) 256
- #define [P3FW_MAX_HAL_RX_BUFFER](#) 256
- #define [P3FW_PCSC_FSDI](#) 5
- #define [P3FW_MAX_ATS_LEN](#) 64
- #define [P3FW_MAX_KEY_LEN](#) 6
- #define [P3FW_MAJOR](#) 0x02U
- #define [P3FW_MINOR](#) 0x02U
- #define [P3FW_BUILD](#) 0x07U
- #define [P3FW_CFG_MAX_SLOTS](#) 0xCA000001U
Configuration options.
- #define [P3FW_CFG_CONT_TIMING](#) 0xCB000001U
- #define [P3FW_CFG_GET_CONT_TIMING](#) 0xCC000001U
- #define [P3FW_CFG_SET_DIP_SWITCHES](#) 0xCD000001U
- #define [P3FW_CFG_BOOTLOADER_VERSION](#) 0xEF000001U
- #define [P3FW_CFG_BOOTLOADER_ACTIVE](#) 0xEF000002U
- #define [P3FW_ERR_AND_NFO_LOOP_DONE](#) 0x00000001U
Error and notifications codes for err_and_nfo_mode.
- #define [P3FW_ERR_AND_NFO_LOOP_UNKNOWN_ERROR](#) 0x00000002U
- #define [P3FW_ERR_AND_NFO_LOOP_CHIP_NOT_DETECTED](#) 0x00000003U

- #define P3FW_ERR_AND_NFO_LOOP_COM_ERROR 0x00000004U
- #define P3FW_ERR_AND_NFO_LOOP_HAL_CAN_NOT_BE_SET 0x00000005U
- #define P3FW_ERR_AND_NFO_LOOP_HW_ERROR 0x00000006U
- #define P3FW_ERR_AND_NFO_LOOP_OS_ERROR 0x00000007U
- #define P3FW_ERR_AND_NFO_LOOP_ERASE_FAILED 0x00000008U
- #define P3FW_ERR_AND_NFO_LOOP_FLASH_FAILED 0x00000009U
- #define P3FW_ERR_AND_NFO_LOOP_BFL_ERROR 0x0000000AU
- #define P3FW_ERR_AND_NFO_LOOP_USB_ERROR 0x0000000BU
- #define P3FW_KEYSTORE_NUM_KEYS 0x05
- #define P3FW_KEYSTORE_NUM_VERS 0x05
- #define P3FW_USB_INT_IN_EP 0x81
- #define P3FW_USB_BULK_OUT_EP 0x05
- #define P3FW_USB_BULK_IN_EP 0x82
- #define P3FW_PCSC_PROTO_PARAMS_LEN_T0 0x05
- #define P3FW_PCSC_PROTO_PARAMS_LEN_T1 0x07
- #define P3FW_CHECK_STATUS(status, error_code)

Typedefs

- typedef struct _p3fw_cl_slot p3fw_cl_slot
Stores state of a contact less slot.
- typedef struct _p3fw_cc_slot p3fw_cc_slot

Enumerations

- enum p3fw_pcsc_mode { P3FW_PCSC_MODE_NORMAL, P3FW_PCSC_MODE_DIRECT }
- enum p3fw_slot_types {
P3FW_SLOT_EMPTY, P3FW_SLOT_SAM, P3FW_SLOT_ISO1444L3A_CARD,
P3FW_SLOT_ISO1444L3B_CARD,
P3FW_SLOT_ISO1444L4_CARD }
Slot is occupied with card type of.
- enum p3fw_bal_configuration { P3FW_BAL_UART, P3FW_BAL_SPI, P3FW_BAL_I2C }
Interfaces NXPRdLib BAL can use.
- enum p3fw_sam { P3FW_SAM_NO, P3FW_SAM_NON_X, P3FW_SAM_IN_X }
How Pegoda uses the SAM.
- enum p3fw_mode { P3FW_MODE_PCSC, P3FW_MODE_DEMO, P3FW_MODE_CFG_OVERWRITE, P3FW_MODE_ACTIVATE_BOOTLOADER }
Available reader modes.

- enum `p3fw_external_interface` { `P3FW_EXT_IF_USB`, `P3FW_EXT_IF_RS232`, `P3FW_EXT_IF_RS485`, `P3FW_EXT_IF_ETHERNET` }

Interfaces Pegoda can use.

- enum `p3fw_ic` { `P3FW_IC_RC523` = 0x01U, `P3FW_IC_SAM` = 0x02U, `P3FW_IC_RC663` = 0x03U }

Reader ICs list.

- enum `p3fw_pcsc_protocol` { `P3FW_PCSC_PROTOCOL_T0` = 0x00U, `P3FW_PCSC_PROTOCOL_T1` = 0x01U, `P3FW_PCSC_PROTOCOL_RAW` = 0x02U, `P3FW_PCSC_PROTOCOL_UNKNOWN` = 0xFFU }

Functions

- void `p3fw_invoke_err_and_nfo_mode` (uint32_t dwErrorCode)
Invokes the error and information loop.
- void `p3fw_task_demo_mode` (void *param)
Task for self contained demo mode.
- void `p3fw_task_poll_and_activate` (void *param)
Task polling and activation of cards.
- void `p3fw_task_ccid_execute` (void *param)
Task for bottom half CCID handling.
- void `p3fw_flash_erase_config` (void)
Erases all configuration options saved on flash.
- phStatus_t `p3fw_flash_read_serial` (uint32_t *pSerNum)
Reads the LPC unique four byte serial number.
- phStatus_t `p3fw_flash_get_config` (uint32_t dwCfgId, uint8_t *pBuff)
Returns a configuration value designated with dwCfgId.
- phStatus_t `p3fw_flash_set_config` (uint32_t dwCfgId, uint8_t *pBuff)
Sets the configuration value designated with dwCfgId.
- phStatus_t `p3fw_slots_init` (void)
Initializes the basic BLF structures and slots data.
- void `p3fw_ext_if_init_usb` (void)
Initialize the USB external interface.
- void `p3fw_slot_add_l3a_card` (uint8_t bSCLlotIndex, uint8_t bSak)

Adds a ISO14443 L3A card to the slot /c bCLSlotIndex.

- void [p3fw_slot_add_l3b_card](#) (uint8_t bCLSlotIndex)

Adds a ISO14443 L3B card to the slot /c bCLSlotIndex.

- void [p3fw_slot_add_l4_card](#) (uint8_t bCLSlotIndex, uint8_t bSak)

Adds a ISO14443 L4 card to the slot /c bCLSlotIndex.

- void [p3fw_slot_remove_cl_card](#) (uint8_t bCLSlotIndex)

Removes card from slot /c bCLSlotIndex.

- phStatus_t [p3fw_slot_get_atr](#) (uint8_t bCCIDSlotIndex, uint8_t *pbAtrBuffer, uint8_t *pbMaxLength)

Returns ATR for a slot /c bCCIDSlotIndex.

- void [p3fw_slot_reset_all_slots](#) (uint8_t contact_card)

Resets all slots to init state.

- void [p3fw_timing_init](#) (void)

Initiates the timing interface.

- void [p3fw_timing_start](#) ()

Resets and starts the timer.

- uint32_t [p3fw_timing_stop](#) ()

Stops the timer and return the elapsed value.

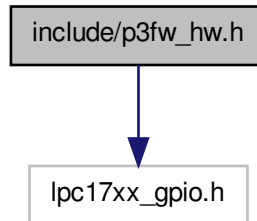
Variables

- [p3fw_data](#) p3fw

6.6 include/p3fw_hw.h File Reference

```
#include <lpc17xx_gpio.h>
```

Include dependency graph for p3fw_hw.h:



Defines

- `#define P3FW_HW_SIGNAL_ANTENNA 0x01`
Signals.
- `#define P3FW_HW_SIGNAL_BEEPER 0x08`
- `#define P3FW_HW_SIGNAL_YELLOW_2 0x10`
- `#define P3FW_HW_SIGNAL_YELLOW_3 0x20`
- `#define P3FW_HW_SIGNAL_YELLOW_4 0x40`

Functions

- `void p3fw_hw_init (void)`
Initialize NXP LPC Hardware.
- `void p3fw_hw_ctrl_signal (uint8_t bSignal, uint8_t bStatus)`
Turn the blue Pegoda 2 antenna on or off.

6.7 include/p3fw_pcsc.h File Reference

Functions

- `phStatus_t p3fw_pcsc_std_ext (uint8_t bCLSlotIndex)`
Checks for and execute a PCSC Standard Extensions.
- `phStatus_t p3fw_pcsc_send_apdu (uint8_t bSW1, uint8_t bSW2, uint16_t wPayloadLength)`

Sends a replay APDU with /c bSW1 and /c bSW2 as SW 1 and SW 2 and payload of length /c wPayloadLength.

- phStatus_t [p3fw_pcsc_std_ext_do_auth](#) (uint8_t bCLSlotIndex, uint8_t bBlock-Number, uint8_t bKeyType)

Authenticates a MIFARE Classic card with key and settings from previous PCSC Standard Extension commands.

- phStatus_t [p3fw_pcsc_std_ext_prepare_l3_card](#) (uint8_t bCLSlotIndex)

Halts and reactivates a L3 card so that it is always in correct state to receive PCSC Standard Extension commands.

6.8 include/p3fw_pins.h File Reference

6.9 include/p3fw_samt1.h File Reference

Data Structures

- struct [sam_t1_param](#)

Defines

- #define [P2_FW_SAM_MODE_PPS](#) 0
- #define [P2_FW_SAM_MODE_X](#) 1
- #define [P2_FW_SAM_FRAME_T1](#) 0
- #define [P2_FW_SAM_FRAME_APDU](#) 1
- #define [P2_FW_TIMING_MODE_NONE](#) 0
- #define [P2_FW_TIMING_MODE_COM](#) 1
- #define [P2_FW_TIMING_MODE_FDT](#) 2
- #define [p3fw_samt1_deinit](#) p2_fw_sam_t1_deinit
- #define [p3fw_samt1_getATR](#) p2_fw_sam_t1_get_atr
- #define [p3fw_samt1_is_sam_inserted](#) p2_fw_sam_t1_is_sam_inserted
- #define [p3fw_samt1_warm_reset](#) p2_fw_sam_t1_warm_reset
- #define [p3fw_samt1_is_busy](#) p2_fw_sam_t1_is_busy
- #define [p3fw_samt1_start](#) p2_fw_sam_t1_start
- #define [p3fw_samt1_send](#) p2_fw_sam_t1_send
- #define [p3fw_samt1_receive](#) p2_fw_sam_t1_receive
- #define [p3fw_samt1_set_etu](#) p2_fw_sam_t1_set_etu
- #define [p3fw_samt1_get_timing_mode](#) p2_fw_sam_t1_get_timing_mode
- #define [p3fw_samt1_set_timing_mode](#) p2_fw_sam_t1_set_timing_mode
- #define [p3fw_samt1_get_cont_tim](#) p2_fw_sam_t1_get_cont_tim

Functions

- void [p2_fw_sam_t1_init](#) (uint8_t chip, uint8_t mode)
- void [p2_fw_sam_t1_deinit](#) (void)
- void [p2_fw_sam_t1_start](#) (void)
- void [p2_fw_sam_t1_send](#) (uint8_t *data, uint32_t len, uint8_t frame_type)
- uint32_t [p2_fw_sam_t1_receive](#) (uint8_t *data, uint8_t frame_type)
- void [p2_fw_sam_t1_get_atr](#) (uint8_t *buffer, uint8_t *max_length)
- void [p2_fw_sam_t1_warm_reset](#) (void)
- Bool [p2_fw_sam_t1_is_busy](#) (void)
- Bool [p2_fw_sam_t1_is_power_off](#) (void)
- Bool [p2_fw_sam_t1_is_sam_inserted](#) (void)
- void [p2_fw_sam_t1_set_rec_extraGuardTime](#) (uint8_t guardTime)
- void [p2_fw_sam_t1_set_bwi_cwi](#) (uint8_t bwi, uint8_t cwi)
- Bool [p2_fw_sam_t1_prepare_pps](#) (uint8_t *ppsData, uint8_t *ppsLen)
- void [p2_fw_sam_t1_pps](#) (void)
- void [p2_fw_sam_t1_set_etu](#) (uint8_t fi_di)
- void [p2_fw_sam_t1_set_my_debug](#) (uint32_t x)
- void [p2_fw_sam_t1_set_timing_mode](#) (uint8_t mode)
- uint8_t [p2_fw_sam_t1_get_timing_mode](#) (void)
- uint32_t [p2_fw_sam_t1_get_cont_tim](#) (void)
- void [EINT3_IRQHandler](#) (void)
- void [TIMER0_IRQHandler](#) (void)
- void [TIMER2_IRQHandler](#) (void)
- phStatus_t [p3fw_samt1_init](#) (enum [p3fw_ic](#) ic, enum [p3fw_sam](#) sam)
- phStatus_t [p3fw_samt1_power_on](#) ()
- void [p3fw_samt1_set_param](#) (uint8_t FI_DI, uint8_t GuardTime, uint8_t BWI_ - CWI, uint8_t ClockStop, uint8_t IFSC)
- phStatus_t [p3fw_samt1_exchange](#) (uint8_t *in, uint32_t in_len, uint8_t *out, uint32_t *out_len)

6.9.1 Define Documentation

6.9.1.1 #define P2_FW_SAM_FRAME_APDU 1

Definition at line 27 of file p3fw_samt1.h.

6.9.1.2 #define P2_FW_SAM_FRAME_T1 0

Definition at line 26 of file p3fw_samt1.h.

6.9.1.3 #define P2_FW_SAM_MODE_PPS 0

Definition at line 23 of file p3fw_samt1.h.

6.9.1.4 #define P2_FW_SAM_MODE_X 1

Definition at line 24 of file p3fw_samt1.h.

6.9.1.5 #define P2_FW_TIMING_MODE_COM 1

Definition at line 30 of file p3fw_samt1.h.

6.9.1.6 #define P2_FW_TIMING_MODE_FDT 2

Definition at line 31 of file p3fw_samt1.h.

6.9.1.7 #define P2_FW_TIMING_MODE_NONE 0

Definition at line 29 of file p3fw_samt1.h.

6.9.1.8 #define p3fw_samt1_deinit p2_fw_sam_t1_deinit

Definition at line 69 of file p3fw_samt1.h.

6.9.1.9 #define p3fw_samt1_get_cont_tim p2_fw_sam_t1_get_cont_tim

Definition at line 80 of file p3fw_samt1.h.

6.9.1.10 #define p3fw_samt1_get_timing_mode p2_fw_sam_t1_get_timing_mode

Definition at line 78 of file p3fw_samt1.h.

6.9.1.11 #define p3fw_samt1_getATR p2_fw_sam_t1_get_atr

Definition at line 70 of file p3fw_samt1.h.

6.9.1.12 #define p3fw_samt1_is_busy p2_fw_sam_t1_is_busy

Definition at line 73 of file p3fw_samt1.h.

6.9.1.13 #define p3fw_samt1_is_sam_inserted p2_fw_sam_t1_is_sam_inserted

Definition at line 71 of file p3fw_samt1.h.

6.9.1.14 #define p3fw_samt1_receive p2_fw_sam_t1_receive

Definition at line 76 of file p3fw_samt1.h.

6.9.1.15 `#define p3fw_samt1_send p2_fw_sam_t1_send`

Definition at line 75 of file p3fw_samt1.h.

6.9.1.16 `#define p3fw_samt1_set_etu p2_fw_sam_t1_set_etu`

Definition at line 77 of file p3fw_samt1.h.

6.9.1.17 `#define p3fw_samt1_set_timing_mode p2_fw_sam_t1_set_timing_mode`

Definition at line 79 of file p3fw_samt1.h.

6.9.1.18 `#define p3fw_samt1_start p2_fw_sam_t1_start`

Definition at line 74 of file p3fw_samt1.h.

6.9.1.19 `#define p3fw_samt1_warm_reset p2_fw_sam_t1_warm_reset`

Definition at line 72 of file p3fw_samt1.h.

6.9.2 Function Documentation

6.9.2.1 `void EINT3_IRQHandler (void)`

Definition at line 815 of file p3fw_samt1.c.

6.9.2.2 `void p2_fw_sam_t1_deinit (void)`

Definition at line 382 of file p3fw_samt1.c.

6.9.2.3 `void p2_fw_sam_t1_get_atr (uint8_t * buffer, uint8_t * max_length)`

Definition at line 717 of file p3fw_samt1.c.

6.9.2.4 `uint32_t p2_fw_sam_t1_get_cont_tim (void)`

Definition at line 156 of file p3fw_samt1.c.

6.9.2.5 `uint8_t p2_fw_sam_t1_get_timing_mode (void)`

Definition at line 151 of file p3fw_samt1.c.

6.9.2.6 void p2_fw_samt1_init (uint8_t *chip*, uint8_t *mode*)

Definition at line 213 of file p3fw_samt1.c.

6.9.2.7 Bool p2_fw_samt1_is_busy (void)

Definition at line 778 of file p3fw_samt1.c.

6.9.2.8 Bool p2_fw_samt1_is_power_off (void)

Definition at line 798 of file p3fw_samt1.c.

6.9.2.9 Bool p2_fw_samt1_is_sam_inserted (void)

Definition at line 808 of file p3fw_samt1.c.

6.9.2.10 void p2_fw_samt1_pps (void)

Definition at line 608 of file p3fw_samt1.c.

6.9.2.11 Bool p2_fw_samt1_prepare_pps (uint8_t * *ppsData*, uint8_t * *ppsLen*)

Definition at line 178 of file p3fw_samt1.c.

6.9.2.12 uint32_t p2_fw_samt1_receive (uint8_t * *data*, uint8_t *frame_type*)

Definition at line 664 of file p3fw_samt1.c.

6.9.2.13 void p2_fw_samt1_send (uint8_t * *data*, uint32_t *len*, uint8_t *frame_type*)

Definition at line 486 of file p3fw_samt1.c.

6.9.2.14 void p2_fw_samt1_set_bwi_cwi (uint8_t *bwi*, uint8_t *cwi*)

Definition at line 174 of file p3fw_samt1.c.

6.9.2.15 void p2_fw_samt1_set_etu (uint8_t *fi_di*)

Definition at line 624 of file p3fw_samt1.c.

6.9.2.16 void p2_fw_samt1_set_my_debug (uint32_t *x*)

Definition at line 140 of file p3fw_samt1.c.

6.9.2.17 void p2_fw_samt1_set_rec_extraGuardTime (uint8_t *guardTime*)

Definition at line 161 of file p3fw_samt1.c.

6.9.2.18 void p2_fw_samt1_set_timing_mode (uint8_t *mode*)

Definition at line 145 of file p3fw_samt1.c.

6.9.2.19 void p2_fw_samt1_start (void)

Definition at line 422 of file p3fw_samt1.c.

6.9.2.20 void p2_fw_samt1_warm_reset (void)

Definition at line 738 of file p3fw_samt1.c.

6.9.2.21 phStatus_t p3fw_samt1_exchange (uint8_t * *in*, uint32_t *in_len*, uint8_t * *out*, uint32_t * *out_len*)

Definition at line 99 of file p3fw_samt1_wrap.c.

6.9.2.22 phStatus_t p3fw_samt1_init (enum p3fw_ic *ic*, enum p3fw_sam *sam*)

Definition at line 23 of file p3fw_samt1_wrap.c.

6.9.2.23 phStatus_t p3fw_samt1_power_on ()

Definition at line 46 of file p3fw_samt1_wrap.c.

6.9.2.24 void p3fw_samt1_set_param (uint8_t *FI_DI*, uint8_t *GuardTime*, uint8_t *BWL_CWL*, uint8_t *ClockStop*, uint8_t *IFSC*)

Definition at line 75 of file p3fw_samt1_wrap.c.

6.9.2.25 void TIMER0_IRQHandler (void)

Definition at line 902 of file p3fw_samt1.c.

6.9.2.26 void TIMER2_IRQHandler (void)

Definition at line 852 of file p3fw_samt1.c.

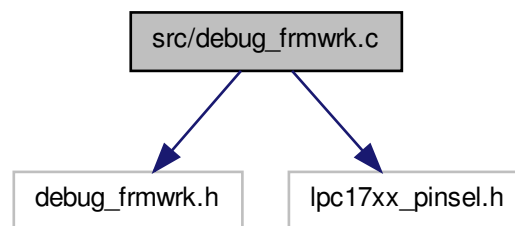
6.10 src/debug_frmwrk.c File Reference

: Contains some utilities that used for debugging through UART

```
#include "debug_frmwrk.h"
```

```
#include "lpc17xx_pinsel.h"
```

Include dependency graph for debug_frmwrk.c:



Defines

- `#define _DEBUG_FRMWRK_`

Functions

- void [UARTPutHexMulti](#) (LPC_UART_TypeDef *UARTx, uint8_t *s, uint32_t len)
Puts a character to UART port.
- void [UARTPutChar](#) (LPC_UART_TypeDef *UARTx, uint8_t ch)
Puts a character to UART port.
- uint8_t [UARTGetChar](#) (LPC_UART_TypeDef *UARTx)
Get a character to UART port.
- void [UARTPuts](#) (LPC_UART_TypeDef *UARTx, const void *str)
Puts a string to UART port.
- void [UARTPuts_](#) (LPC_UART_TypeDef *UARTx, const void *str)
Puts a string to UART port and print new line.
- void [UARTPutDec](#) (LPC_UART_TypeDef *UARTx, uint8_t decnum)
Puts a decimal number to UART port.

- void [UARTPutDec16](#) (LPC_UART_TypeDef *UARTx, uint16_t decnum)
Puts a decimal number to UART port.
- void [UARTPutDec32](#) (LPC_UART_TypeDef *UARTx, uint32_t decnum)
Puts a decimal number to UART port.
- void [UARTPutHex](#) (LPC_UART_TypeDef *UARTx, uint8_t hexnum)
Puts a hex number to UART port.
- void [UARTPutHex16](#) (LPC_UART_TypeDef *UARTx, uint16_t hexnum)
Puts a hex number to UART port.
- void [UARTPutHex32](#) (LPC_UART_TypeDef *UARTx, uint32_t hexnum)
Puts a hex number to UART port.
- void [debug_frmwrk_init](#) (void)
Initialize Debug frame work through initializing UART port.

Variables

- void(* [_db_mhex](#))(LPC_UART_TypeDef *UARTx, uint8_t *s, uint32_t len)
- void(* [_db_msg](#))(LPC_UART_TypeDef *UARTx, const void *s)
- void(* [_db_msg_](#))(LPC_UART_TypeDef *UARTx, const void *s)
- void(* [_db_char](#))(LPC_UART_TypeDef *UARTx, uint8_t ch)
- void(* [_db_dec](#))(LPC_UART_TypeDef *UARTx, uint8_t decn)
- void(* [_db_dec_16](#))(LPC_UART_TypeDef *UARTx, uint16_t decn)
- void(* [_db_dec_32](#))(LPC_UART_TypeDef *UARTx, uint32_t decn)
- void(* [_db_hex](#))(LPC_UART_TypeDef *UARTx, uint8_t hexn)
- void(* [_db_hex_16](#))(LPC_UART_TypeDef *UARTx, uint16_t hexn)
- void(* [_db_hex_32](#))(LPC_UART_TypeDef *UARTx, uint32_t hexn)
- uint8_t(* [_db_get_char](#))(LPC_UART_TypeDef *UARTx)

6.10.1 Detailed Description

: Contains some utilities that used for debugging through UART

Version

: 1.0

Date

: 18. Mar. 2009

Author

: HieuNguyen ----- Software that is described herein is for illustrative purposes only which provides customers with programming information regarding the products. This software is supplied "AS IS" without any warranties. NXP Semiconductors assumes no responsibility or liability for the use of the software, conveys no license or title under any patent, copyright, or mask work right to the product. NXP Semiconductors reserves the right to make changes in the software without notification. NXP Semiconductors also make no representation or warranty that such application will be suitable for the specified use without further testing or modification.

Definition in file [debug_frmwrk.c](#).

6.10.2 Define Documentation**6.10.2.1 #define _DEBUG_FRMWRK_**

Definition at line 21 of file debug_frmwrk.c.

6.10.3 Function Documentation**6.10.3.1 void debug_frmwrk_init (void)**

Initialize Debug frame work through initializing UART port.

Parameters

in	None	
----	------	--

Returns

None

Definition at line 242 of file debug_frmwrk.c.

6.10.3.2 uint8_t UARTGetChar (LPC_UART_TypeDef * UARTx)

Get a character to UART port.

Parameters

in	UARTx	Pointer to UART peripheral
in	ch	Character to put

Returns

None

Definition at line 77 of file debug_frmwrk.c.

6.10.3.3 void UARTPutChar (LPC_UART_TypeDef * *UARTx*, uint8_t *ch*)

Puts a character to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>ch</i>	Character to put

Returns

None

Definition at line 65 of file debug_frmwrk.c.

6.10.3.4 void UARTPutDec (LPC_UART_TypeDef * *UARTx*, uint8_t *decnum*)

Puts a decimal number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>decnum</i>	Decimal number (8-bit long)

Returns

None

Definition at line 121 of file debug_frmwrk.c.

6.10.3.5 void UARTPutDec16 (LPC_UART_TypeDef * *UARTx*, uint16_t *decnum*)

Puts a decimal number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>decnum</i>	Decimal number (8-bit long)

Returns

None

Definition at line 137 of file debug_frmwrk.c.

6.10.3.6 void UARTPutDec32 (LPC_UART_TypeDef * *UARTx*, uint32_t *decnum*)

Puts a decimal number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>decnum</i>	Decimal number (8-bit long)

Returns

None

Definition at line 157 of file debug_frmwrk.c.

6.10.3.7 void UARTPutHex (LPC_UART_TypeDef * *UARTx*, uint8_t *hexnum*)

Puts a hex number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>hexnum</i>	Hex number (8-bit long)

Returns

None

Definition at line 187 of file debug_frmwrk.c.

6.10.3.8 void UARTPutHex16 (LPC_UART_TypeDef * *UARTx*, uint16_t *hexnum*)

Puts a hex number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>hexnum</i>	Hex number (16-bit long)

Returns

None

Definition at line 206 of file debug_frmwrk.c.

6.10.3.9 void UARTPutHex32 (LPC_UART_TypeDef * *UARTx*, uint32_t *hexnum*)

Puts a hex number to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>hexnum</i>	Hex number (32-bit long)

Returns

None

Definition at line 224 of file debug_frmwrk.c.

6.10.3.10 void UARTPutHexMulti (LPC_UART_TypeDef * *UARTx*, uint8_t * *s*, uint32_t *len*)

Definition at line 40 of file debug_frmwrk.c.

6.10.3.11 void UARTPuts (LPC_UART_TypeDef * *UARTx*, const void * *str*)

Puts a string to UART port.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>str</i>	string to put

Returns

None

Definition at line 91 of file debug_frmwrk.c.

6.10.3.12 void UARTPuts_ (LPC_UART_TypeDef * *UARTx*, const void * *str*)

Puts a string to UART port and print new line.

Parameters

in	<i>UARTx</i>	Pointer to UART peripheral
in	<i>str</i>	String to put

Returns

None

Definition at line 108 of file debug_frmwrk.c.

6.10.4 Variable Documentation

6.10.4.1 void(* _db_char)(LPC_UART_TypeDef *UARTx, uint8_t ch)

Definition at line 31 of file debug_frmwrk.c.

6.10.4.2 void(* _db_dec)(LPC_UART_TypeDef *UARTx, uint8_t decn)

Definition at line 32 of file debug_frmwrk.c.

6.10.4.3 void(* _db_dec_16)(LPC_UART_TypeDef *UARTx, uint16_t decn)

Definition at line 33 of file debug_frmwrk.c.

6.10.4.4 void(* _db_dec_32)(LPC_UART_TypeDef *UARTx, uint32_t decn)

Definition at line 34 of file debug_frmwrk.c.

6.10.4.5 uint8_t(* _db_get_char)(LPC_UART_TypeDef *UARTx)

Definition at line 38 of file debug_frmwrk.c.

6.10.4.6 void(* _db_hex)(LPC_UART_TypeDef *UARTx, uint8_t hexn)

Definition at line 35 of file debug_frmwrk.c.

6.10.4.7 void(* _db_hex_16)(LPC_UART_TypeDef *UARTx, uint16_t hexn)

Definition at line 36 of file debug_frmwrk.c.

6.10.4.8 void(* _db_hex_32)(LPC_UART_TypeDef *UARTx, uint32_t hexn)

Definition at line 37 of file debug_frmwrk.c.

6.10.4.9 void(* _db_mhex)(LPC_UART_TypeDef *UARTx, uint8_t *s, uint32_t len)

Definition at line 28 of file debug_frmwrk.c.

6.10.4.10 void(* _db_msg)(LPC_UART_TypeDef *UARTx, const void *s)

Definition at line 29 of file debug_frmwrk.c.

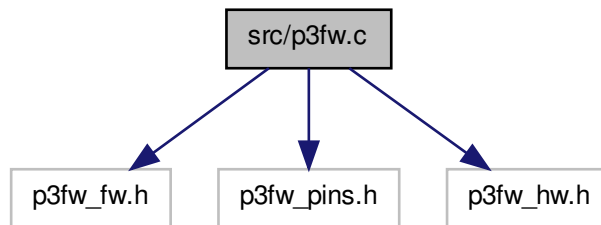
6.10.4.11 void(*_db_msg_)(LPC_UART_TypeDef *UARTx, const void *s)

Definition at line 30 of file debug_frmwrk.c.

6.11 src/p3fw.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_pins.h>
#include <p3fw_hw.h>
```

Include dependency graph for p3fw.c:



Defines

- #define P3FW_DIP_CFG_BAL_SPI 0x10U
- #define P3FW_DIP_CFG_BAL_I2C 0x00U
- #define P3FW_DIP_CFG_BAL_UART 0x20U
- #define P3FW_DIP_CFG_SAM_NO 0x00U
- #define P3FW_DIP_CFG_SAM_NON_X 0x01U
- #define P3FW_DIP_CFG_SAM_IN_X 0x02U
- #define P3FW_DIP_MODE_PCSC 0x00U
- #define P3FW_DIP_MODE_DEMO 0x40U
- #define P3FW_DIP_MODE_CFG_OVERWRITE 0x80U
- #define P3FW_DIP_MODE_ACT_BOOTLOADER 0xC0U
- #define P3FW_DIP_EXT_IF_USB 0x00U
- #define P3FW_DIP_EXT_IF_RS232 0x04U
- #define P3FW_DIP_EXT_IF_RS485 0x08U
- #define P3FW_DIP_EXT_IF_ETHERNET 0x0CU
- #define P3FW_TASK_START_EXECUTE 0x01U
- #define P3FW_TASK_START_POLL 0x02U
- #define P3FW_TASK_START_ERROR 0x10U

Functions

- int [main](#) ()
- void [p3fw_dump_regs](#) ()

Variables

- [p3fw_data](#) p3fw

6.11.1 Define Documentation

6.11.1.1 `#define P3FW_DIP_CFG_BAL_I2C 0x00U`

Definition at line 24 of file p3fw.c.

6.11.1.2 `#define P3FW_DIP_CFG_BAL_SPI 0x10U`

Definition at line 23 of file p3fw.c.

6.11.1.3 `#define P3FW_DIP_CFG_BAL_UART 0x20U`

Definition at line 25 of file p3fw.c.

6.11.1.4 `#define P3FW_DIP_CFG_SAM_IN_X 0x02U`

Definition at line 29 of file p3fw.c.

6.11.1.5 `#define P3FW_DIP_CFG_SAM_NO 0x00U`

Definition at line 27 of file p3fw.c.

6.11.1.6 `#define P3FW_DIP_CFG_SAM_NON_X 0x01U`

Definition at line 28 of file p3fw.c.

6.11.1.7 `#define P3FW_DIP_EXT_IF_ETHERNET 0x0CU`

Definition at line 39 of file p3fw.c.

6.11.1.8 `#define P3FW_DIP_EXT_IF_RS232 0x04U`

Definition at line 37 of file p3fw.c.

6.11.1.9 #define P3FW_DIP_EXT_IF_RS485 0x08U

Definition at line 38 of file p3fw.c.

6.11.1.10 #define P3FW_DIP_EXT_IF_USB 0x00U

Definition at line 36 of file p3fw.c.

6.11.1.11 #define P3FW_DIP_MODE_ACT_BOOTLOADER 0xC0U

Definition at line 34 of file p3fw.c.

6.11.1.12 #define P3FW_DIP_MODE_CFG_OVERWRITE 0x80U

Definition at line 33 of file p3fw.c.

6.11.1.13 #define P3FW_DIP_MODE_DEMO 0x40U

Definition at line 32 of file p3fw.c.

6.11.1.14 #define P3FW_DIP_MODE_PCSC 0x00U

Definition at line 31 of file p3fw.c.

6.11.1.15 #define P3FW_TASK_START_ERROR 0x10U

Definition at line 43 of file p3fw.c.

6.11.1.16 #define P3FW_TASK_START_EXECUTE 0x01U

Definition at line 41 of file p3fw.c.

6.11.1.17 #define P3FW_TASK_START_POLL 0x02U

Definition at line 42 of file p3fw.c.

6.11.2 Function Documentation**6.11.2.1 int main ()**

Definition at line 49 of file p3fw.c.

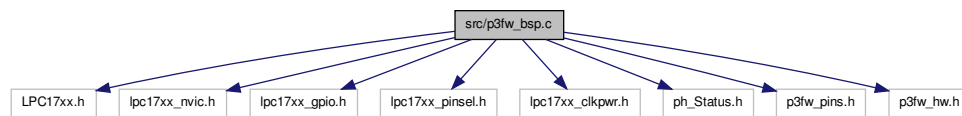
6.11.2.2 void p3fw_dump_regs ()

Definition at line 430 of file p3fw.c.

6.12 src/p3fw_bsp.c File Reference

```
#include <LPC17xx.h>
#include <lpc17xx_nvic.h>
#include <lpc17xx_gpio.h>
#include <lpc17xx_pinsel.h>
#include <lpc17xx_clkpwr.h>
#include <ph_Status.h>
#include <p3fw_pins.h>
#include <p3fw_hw.h>
```

Include dependency graph for p3fw_bsp.c:



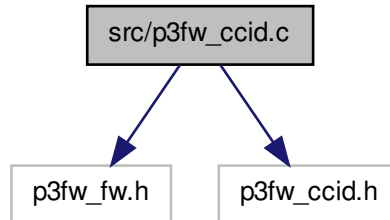
Functions

- void [p3fw_hw_init](#) (void)
Initialize NXP LPC Hardware.
- void [p3fw_hw_ctrl_signal](#) (uint8_t bSignal, uint8_t bStatus)
Turn the blue Pegoda 2 antenna on or off.

6.13 src/p3fw_ccid.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_ccid.h>
```

Include dependency graph for p3fw_ccid.c:



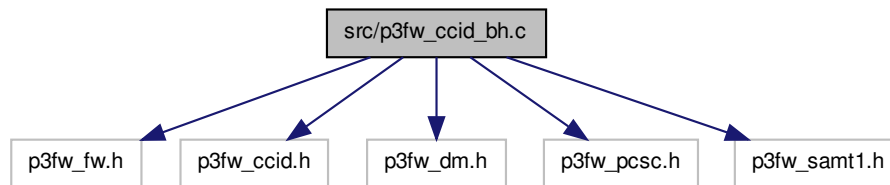
Functions

- void [p3fw_task_ccid_execute](#) (void *param)
Task for bottom half CCID handling.
- void [p3fw_ccid_send_notify](#) ()
Sends slot status notification through INT end-point.
- void [p3fw_ccid_send_frame](#) (uint8_t bMessageType, uint8_t bByte1, uint8_t bByte2, uint8_t bByte3)
Sends CCID frame over USB to the driver.
- uint8_t [p3fw_ccid_input_header_get_byte](#) (uint8_t bByte)
Gets one byte from input header.
- uint8_t * [p3fw_ccid_get_output_payload_buffer](#) ()
Returns a pointer to CCID internal output buffer.
- void [p3fw_ccid_set_output_payload_length](#) (uint32_t dwPayloadLength)
Sets the payload length of the output CCID message.
- uint8_t * [p3fw_ccid_get_input_payload_buffer](#) ()
Returns a pointer to CCID internal input buffer.
- uint32_t [p3fw_ccid_get_input_payload_length](#) ()
Gets the payload length of the input CCID message.

6.14 src/p3fw_ccid_bh.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_ccid.h>
#include <p3fw_dm.h>
#include <p3fw_pcsc.h>
#include <p3fw_samt1.h>
```

Include dependency graph for p3fw_ccid_bh.c:



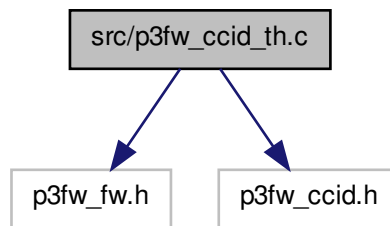
Functions

- void [p3fw_ccid_bh_get_slot_status](#) (uint8_t bCCIDSlotIndex)
Returns /c bCCIDSlotIndex status.
- void [p3fw_ccid_bh_icc_power_on](#) (uint8_t bCCIDSlotIndex)
Card Power On.
- void [p3fw_ccid_bh_icc_power_off](#) (uint8_t bCCIDSlotIndex)
Card Power Off.
- void [p3fw_ccid_bh_get_parameters](#) (uint8_t bCCIDSlotIndex)
CCID Get Parameters.
- void [p3fw_ccid_bh_set_parameters](#) (uint8_t bCCIDSlotIndex)
CCID Set Parameters.
- void [p3fw_ccid_bh_escape](#) (uint8_t bCCIDSlotIndex)
CCID Escape Interface (SCardControl)
- void [p3fw_ccid_bh_xfer](#) (uint8_t bCCIDSlotIndex)
CCID Transfer command.

6.15 src/p3fw_ccid_th.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_ccid.h>
```

Include dependency graph for p3fw_ccid_th.c:



Functions

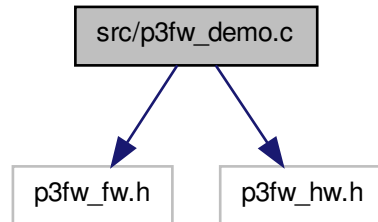
- void [p3fw_ccid_th_dispatch](#) (void)

Main function for top half CCID handler.

6.16 src/p3fw_demo.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_hw.h>
```

Include dependency graph for p3fw_demo.c:



Defines

- `#define P3FW_DEMO_TX_BUFFER 1024`
- `#define P3FW_DEMO_RX_BUFFER 1024`
- `#define P3FW_DEMO_ATQB_LEN 14`
- `#define P3FW_DEMO_UID_LEN 10`

Functions

- `void p3fw_task_demo_mode (void *param)`
Task for self contained demo mode.

6.16.1 Define Documentation

6.16.1.1 `#define P3FW_DEMO_ATQB_LEN 14`

Definition at line 41 of file `p3fw_demo.c`.

6.16.1.2 `#define P3FW_DEMO_RX_BUFFER 1024`

Definition at line 38 of file `p3fw_demo.c`.

6.16.1.3 `#define P3FW_DEMO_TX_BUFFER 1024`

Definition at line 37 of file `p3fw_demo.c`.

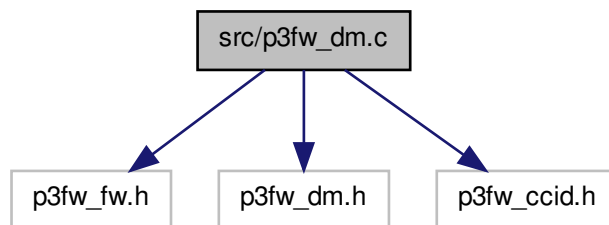
6.16.1.4 #define P3FW_DEMO_UID_LEN 10

Definition at line 44 of file p3fw_demo.c.

6.17 src/p3fw_dm.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_dm.h>
#include <p3fw_ccid.h>
```

Include dependency graph for p3fw_dm.c:



Functions

- void [p3fw_dm_xfer\(\)](#)
Direct Mode Transfer Mode.
- uint8_t [p3fw_dm_get_class\(\)](#)
Extracts class from the frame.
- uint8_t [p3fw_dm_get_instruction\(\)](#)
Extracts instruction from the frame.
- uint16_t [p3fw_dm_get_input_payload_length\(\)](#)
Gets input length.
- uint8_t * [p3fw_dm_get_input_payload\(\)](#)
Gets input payload buffer.
- uint8_t * [p3fw_dm_get_output_payload\(\)](#)

Gets output payload buffer.

- uint8_t * [p3fw_dm_get_output_pointer_uint8_t](#) (uint16_t wDmByteIndex)
- uint8_t [p3fw_dm_get_slot_index](#) ()

Extracts slot index from the frame.

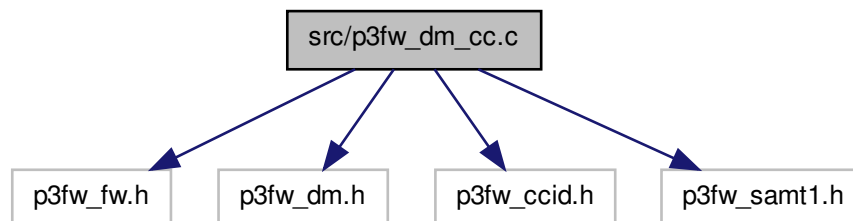
- uint8_t [p3fw_dm_get_input_uint8_t](#) (uint16_t wDmByteIndex)
- uint8_t * [p3fw_dm_get_input_pointer_uint8_t](#) (uint16_t wDmByteIndex)
- uint16_t [p3fw_dm_get_input_uint16_t](#) (uint16_t wDmByteIndex)
- uint32_t [p3fw_dm_get_input_uint32_t](#) (uint16_t wDmByteIndex)
- void [p3fw_dm_send_frame](#) (phStatus_t status, uint16_t wFrameLength)

Sends frame.

6.18 src/p3fw_dm_cc.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_dm.h>
#include <p3fw_ccid.h>
#include <p3fw_samt1.h>
```

Include dependency graph for p3fw_dm_cc.c:



Functions

- void [p3fw_dm_contact_card](#) ()

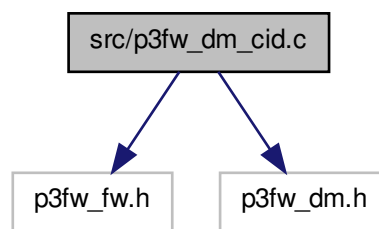
Direct Mode Contact Card Functions.

6.19 src/p3fw_dm_cid.c File Reference

```
#include <p3fw_fw.h>
```

```
#include <p3fw_dm.h>
```

Include dependency graph for p3fw_dm_cid.c:



Functions

- void [p3fw_dm_cid](#) ()

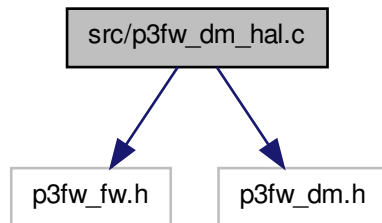
Direct Mode CID Management Functions.

6.20 src/p3fw_dm_hal.c File Reference

```
#include <p3fw_fw.h>
```

```
#include <p3fw_dm.h>
```

Include dependency graph for p3fw_dm_hal.c:



Functions

- void [p3fw_dm_hal\(\)](#)

Direct Mode Hardware Abstraction Layer Functions.

Variables

- struct [p3fw_hal_buffer](#) `aHalBuffers` [P3FW_MAX_CONTACTLESS_SLOTS]

6.20.1 Variable Documentation

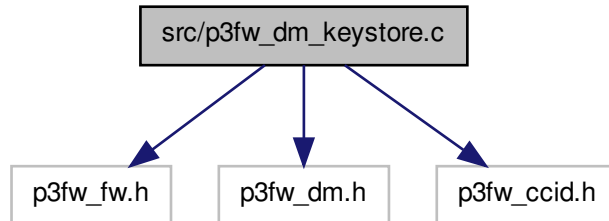
6.20.1.1 struct `p3fw_hal_buffer aHalBuffers`[P3FW_MAX_CONTACTLESS_SLOTS]

Definition at line 25 of file `p3fw_slots.c`.

6.21 src/p3fw_dm_keystore.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_dm.h>
#include <p3fw_ccid.h>
```

Include dependency graph for p3fw_dm_keystore.c:



Functions

- void [p3fw_dm_keystore](#) ()

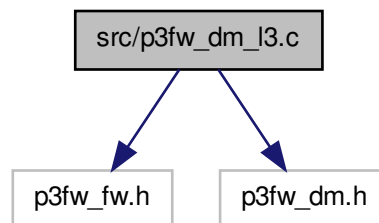
Direct Mode Keystore Management Functions.

6.22 src/p3fw_dm_l3.c File Reference

```
#include <p3fw_fw.h>
```

```
#include <p3fw_dm.h>
```

Include dependency graph for p3fw_dm_l3.c:



Functions

- void [p3fw_dm_l3](#) ()

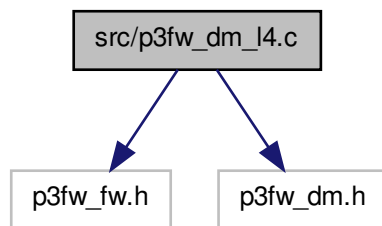
Direct Mode ISO14443-3 Functions.

6.23 src/p3fw_dm_l4.c File Reference

```
#include <p3fw_fw.h>
```

```
#include <p3fw_dm.h>
```

Include dependency graph for p3fw_dm_l4.c:



Functions

- void [p3fw_dm_l4](#) ()

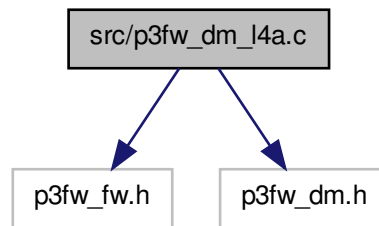
Direct Mode ISO14443-4 Functions.

6.24 src/p3fw_dm_l4a.c File Reference

```
#include <p3fw_fw.h>
```

```
#include <p3fw_dm.h>
```

Include dependency graph for p3fw_dm_l4a.c:



Functions

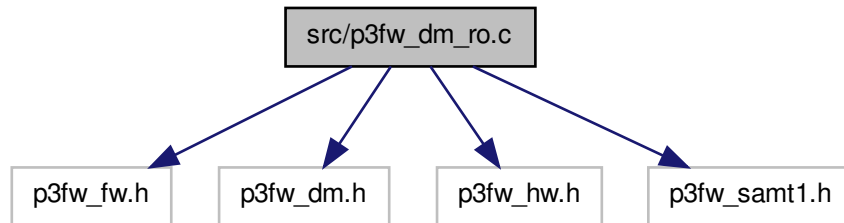
- void `p3fw_dm_l4a` ()

Direct Mode ISO14443-4A Functions.

6.25 src/p3fw_dm_ro.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_dm.h>
#include <p3fw_hw.h>
#include <p3fw_samt1.h>
```

Include dependency graph for p3fw_dm_ro.c:



Functions

- void [p3fw_dm_ro](#) ()

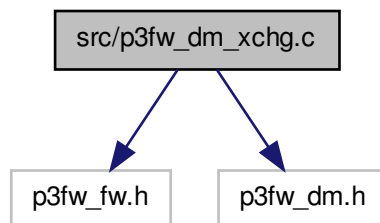
Direct Mode Reader Operations (also available through SCardControl/CCID Escape Interface)

6.26 src/p3fw_dm_xchg.c File Reference

```
#include <p3fw_fw.h>
```

```
#include <p3fw_dm.h>
```

Include dependency graph for p3fw_dm_xchg.c:



Functions

- void [p3fw_dm_xchg](#) ()

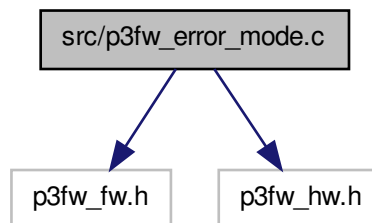
Direct Mode Exchange Functions.

6.27 src/p3fw_error_mode.c File Reference

```
#include <p3fw_fw.h>
```

```
#include <p3fw_hw.h>
```

Include dependency graph for p3fw_error_mode.c:



Functions

- void [p3fw_invoke_err_and_nfo_mode](#) (uint32_t dwErrorCode)

Invokes the error and information loop.

6.28 src/p3fw_ext_if_usb.c File Reference

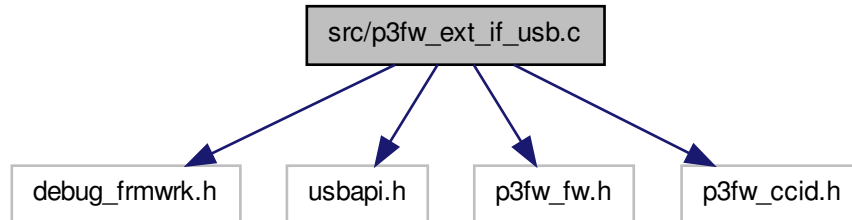
```
#include <debug_frmwrk.h>
```

```
#include <usbapi.h>
```

```
#include <p3fw_fw.h>
```

```
#include <p3fw_ccid.h>
```

Include dependency graph for p3fw_ext_if_usb.c:



Defines

- `#define P3FW_USB_LE_WORD(x) ((x)&0xFF),((x)>>8)`
- `#define P3FW_USB_LE_DWORD(x) ((x)&0xFF),(((x)>>8)&0xFF),(((x)>>16)&0xFF),((x)>>24)`
- `#define P3FW_USB_MAX_PACKET_SIZE 64`
- `#define P3FW_USB_CONTROL_REQUEST_ABORT 0x01`
- `#define P3FW_USB_CONTROL_REQUEST_GET_CLOCK_FREQUENCIES 0x02`
- `#define P3FW_USB_CONTROL_REQUEST_GET_DATA_RATES 0x03`
- `#define P3FW_SERNUM_USB_OFFSET 147`
- `#define P3FW_NAME_USB_OFFSET 143`
- `#define P3FW_MAX_SLOT_INDEX 40`

Functions

- `void p3fw_ext_if_init_usb (void)`
Initialize the USB external interface.

6.28.1 Define Documentation

6.28.1.1 `#define P3FW_MAX_SLOT_INDEX 40`

Definition at line 37 of file p3fw_ext_if_usb.c.

6.28.1.2 `#define P3FW_NAME_USB_OFFSET 143`

Definition at line 36 of file p3fw_ext_if_usb.c.

6.28.1.3 #define P3FW_SERNUM_USB_OFFSET 147

Definition at line 35 of file p3fw_ext_if_usb.c.

6.28.1.4 #define P3FW_USB_CONTROL_REQUEST_ABORT 0x01

Definition at line 31 of file p3fw_ext_if_usb.c.

6.28.1.5 #define P3FW_USB_CONTROL_REQUEST_GET_CLOCK_FREQUENCIES 0x02

Definition at line 32 of file p3fw_ext_if_usb.c.

6.28.1.6 #define P3FW_USB_CONTROL_REQUEST_GET_DATA_RATES 0x03

Definition at line 33 of file p3fw_ext_if_usb.c.

**6.28.1.7 #define P3FW_USB_LE_DWORD(x
) ((x)&0xFF),(((x)>>8)&0xFF),(((x)>>16)&0xFF),((x)>>24)**

Definition at line 27 of file p3fw_ext_if_usb.c.

6.28.1.8 #define P3FW_USB_LE_WORD(x) ((x)&0xFF),((x)>>8)

Definition at line 26 of file p3fw_ext_if_usb.c.

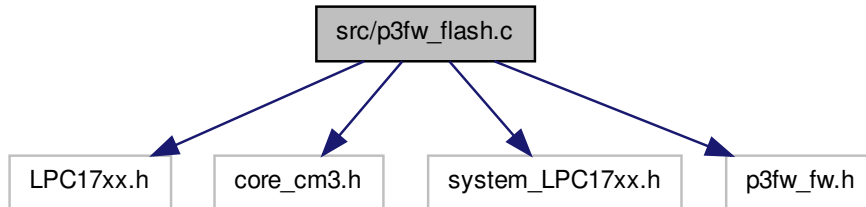
6.28.1.9 #define P3FW_USB_MAX_PACKET_SIZE 64

Definition at line 29 of file p3fw_ext_if_usb.c.

6.29 src/p3fw_flash.c File Reference

```
#include <LPC17xx.h>
#include <core_cm3.h>
#include <system_LPC17xx.h>
#include <p3fw_fw.h>
```

Include dependency graph for p3fw_flash.c:



Defines

- #define [IAP_LOCATION](#) 0x1FFF1FF1
- #define [IAP_SEC_29](#) 0x00078000
- #define [IAP_PREP_SECTORS](#) 50
- #define [IAP_COPY_RAM_TO_FLASH](#) 51
- #define [IAP_ERASE_SECTOR](#) 52
- #define [IAP_BLANK_CHECK_SECTOR](#) 53
- #define [IAP_READ_ID](#) 54
- #define [IAP_READ_BOOT_CODE_VER](#) 55
- #define [IAP_READ_DEV_SER_NUM](#) 58
- #define [IAP_COMPARE](#) 56
- #define [IAP_REINVOKE](#) 57
- #define [P3FW_CFG_S1](#) 0x00
- #define [P3FW_CFG_S2](#) 0x01
- #define [P3FW_CFG_COUNT](#) 0x02
- #define [P3FW_CFG_ENTRY_OFFSET](#) 0x04
- #define [P3FW_CFG_ENTRY_SIZE](#) 0x24
- #define [P3FW_CFG_MAX](#) 14
- #define [P3FW_FLASH_BUFFER](#) 256

Typedefs

- typedef void(* [IAP](#))(long[], long[])

Functions

- void [p3fw_flash_erase_config](#) (void)
Erases all configuration options saved on flash.

- `phStatus_t p3fw_flash_read_serial (uint32_t *pSerNum)`
Reads the LPC unique four byte serial number.
- `phStatus_t p3fw_flash_get_config (uint32_t dwCfgId, uint8_t *pBuff)`
Returns a configuration value designated with `dwCfgId`.
- `phStatus_t p3fw_flash_set_config (uint32_t dwCfgId, uint8_t *pBuff)`
Sets the configuration value designated with `dwCfgId`.

6.29.1 Define Documentation

6.29.1.1 `#define IAP_BLANK_CHECK_SECTOR` 53

Definition at line 32 of file `p3fw_flash.c`.

6.29.1.2 `#define IAP_COMPARE` 56

Definition at line 36 of file `p3fw_flash.c`.

6.29.1.3 `#define IAP_COPY_RAM_TO_FLASH` 51

Definition at line 30 of file `p3fw_flash.c`.

6.29.1.4 `#define IAP_ERASE_SECTOR` 52

Definition at line 31 of file `p3fw_flash.c`.

6.29.1.5 `#define IAP_LOCATION` 0xFF1FF1

Definition at line 25 of file `p3fw_flash.c`.

6.29.1.6 `#define IAP_PREP_SECTORS` 50

Definition at line 29 of file `p3fw_flash.c`.

6.29.1.7 `#define IAP_READ_BOOT_CODE_VER` 55

Definition at line 34 of file `p3fw_flash.c`.

6.29.1.8 `#define IAP_READ_DEV_SER_NUM` 58

Definition at line 35 of file `p3fw_flash.c`.

6.29.1.9 #define IAP_READ_ID 54

Definition at line 33 of file p3fw_flash.c.

6.29.1.10 #define IAP_REINVOKE 57

Definition at line 37 of file p3fw_flash.c.

6.29.1.11 #define IAP_SEC_29 0x00078000

Definition at line 27 of file p3fw_flash.c.

6.29.1.12 #define P3FW_CFG_COUNT 0x02

Definition at line 41 of file p3fw_flash.c.

6.29.1.13 #define P3FW_CFG_ENTRY_OFFSET 0x04

Definition at line 42 of file p3fw_flash.c.

6.29.1.14 #define P3FW_CFG_ENTRY_SIZE 0x24

Definition at line 43 of file p3fw_flash.c.

6.29.1.15 #define P3FW_CFG_MAX 14

Definition at line 45 of file p3fw_flash.c.

6.29.1.16 #define P3FW_CFG_S1 0x00

Definition at line 39 of file p3fw_flash.c.

6.29.1.17 #define P3FW_CFG_S2 0x01

Definition at line 40 of file p3fw_flash.c.

6.29.1.18 #define P3FW_FLASH_BUFFER 256

Definition at line 47 of file p3fw_flash.c.

6.29.2 Typedef Documentation

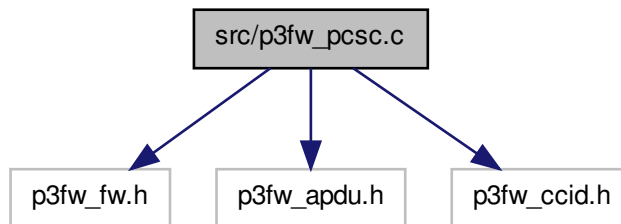
6.29.2.1 typedef void(* IAP)(long[], long[])

Definition at line 49 of file p3fw_flash.c.

6.30 src/p3fw_pcsc.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_apdu.h>
#include <p3fw_ccid.h>
```

Include dependency graph for p3fw_pcsc.c:



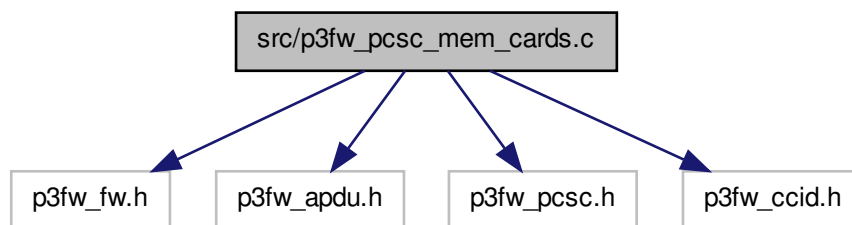
Functions

- phStatus_t [p3fw_pcsc_send_apdu](#) (uint8_t bSW1, uint8_t bSW2, uint16_t wPayloadLength)
Sends a replay APDU with /c bSW1 and /c bSW2 as SW 1 and SW 2 and payload of length /c wPayloadLength.
- phStatus_t [p3fw_pcsc_std_ext_do_auth](#) (uint8_t bCLSlotIndex, uint8_t bBlockNumber, uint8_t bKeyType)
Authenticates a MIFARE Classic card with key and settings from previous PCSC Standard Extension commands.
- phStatus_t [p3fw_pcsc_std_ext_prepare_l3_card](#) (uint8_t bCLSlotIndex)
Halts and reactivates a L3 card so that it is always in correct state to receive PCSC Standard Extension commands.

6.31 src/p3fw_pcsc_mem_cards.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_apdu.h>
#include <p3fw_pcsc.h>
#include <p3fw_ccid.h>
```

Include dependency graph for p3fw_pcsc_mem_cards.c:



Functions

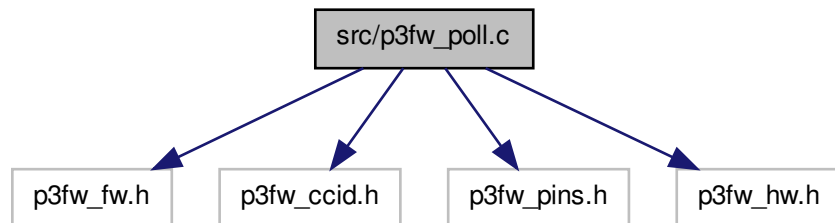
- phStatus_t [p3fw_pcsc_std_ext](#) (uint8_t bCLSlotIndex)

Checks for and execute a PCSC Standard Extensions.

6.32 src/p3fw_poll.c File Reference

```
#include <p3fw_fw.h>
#include <p3fw_ccid.h>
#include <p3fw_pins.h>
#include <p3fw_hw.h>
```

Include dependency graph for p3fw_poll.c:



Functions

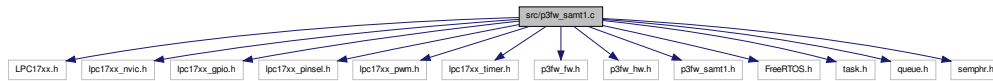
- void [p3fw_task_poll_and_activate](#) (void *param)

Task polling and activation of cards.

6.33 src/p3fw_samt1.c File Reference

```
#include <LPC17xx.h>
#include <lpc17xx_nvic.h>
#include <lpc17xx_gpio.h>
#include <lpc17xx_pinsel.h>
#include <lpc17xx_pwm.h>
#include <lpc17xx_timer.h>
#include <p3fw_fw.h>
#include <p3fw_hw.h>
#include <p3fw_samt1.h>
#include <FreeRTOS.h>
#include <task.h>
#include <queue.h>
#include <semphr.h>
```

Include dependency graph for p3fw_samt1.c:



Data Structures

- struct [p2_fw_SAM_ctrl_](#)
- struct [cpot_atr_frame](#)

Defines

- #define [P2_FW_SAM_DEBUG](#) 0
- #define [P2_FW_SAM_T1_CLOCK_CONFIG](#) 1
- #define [T1_BYTE_WAIT_TIME](#) 5
- #define [P2_FW_SAM_PWM_CH0_MATCH_VALUE](#) 4
- #define [P2_FW_SAM_PWM_CH1_MATCH_VALUE](#) 2
- #define [P2_FW_SAM_TIMER_PRESCALE_VALUE](#) 4
- #define [P2_FW_SAM_CLOCK_LEN_NS](#) 222
- #define [P2_FW_SAM_MODE_RECEIVE](#) 0
- #define [P2_FW_SAM_MODE_SEND](#) 1
- #define [P2_FW_SAM_MODE_RESET](#) 2
- #define [P2_FW_SAM_MODE_IDLE](#) 4
- #define [P2_FW_SAM_MODE_WAIT_START_BIT](#) 6
- #define [P2_FW_SAM_MODE_POWER_OFF](#) 7
- #define [BITBAND_PERI_REF](#) 0x40000000
- #define [BITBAND_PERI_BASE](#) 0x42000000
- #define [BITBAND_PERI\(a, b\)](#) ((BITBAND_PERI_BASE + ((a)-BITBAND_PERI_REF)*32 + ((b)*4)))
- #define [TIMER0_IR](#) 0x40004000
- #define [TIMER0_TCR](#) 0x40004004
- #define [TIMER0_IR_MR0](#) *((volatile unsigned char *) (BITBAND_PERI(TIMER0_IR,0)))
- #define [TIMER0_TCR_ENABLE](#) *((volatile unsigned char *) (BITBAND_PERI(TIMER0_TCR,0)))
- #define [TIMER0_TCR_RESET](#) *((volatile unsigned char *) (BITBAND_PERI(TIMER0_TCR,1)))
- #define [P2_FW_SAM_CONVERSION_UNKNOWN](#) 0
- #define [P2_FW_SAM_CONVERSION_DIRECT](#) 1
- #define [P2_FW_SAM_CONVERSION_INVERSE](#) 2
- #define [P2_FW_SAM_MAX_ATR_SIZE](#) 33
- #define [P2_FW_SAM_MAX_REC_DATALEN](#) 600
- #define [P2_FW_SAM_DEFAULT_BWI](#) 8405405

Functions

- void [p2_fw_sam_t1_set_my_debug](#) (uint32_t x)
- void [p2_fw_sam_t1_set_timing_mode](#) (uint8_t mode)
- uint8_t [p2_fw_sam_t1_get_timing_mode](#) (void)
- uint32_t [p2_fw_sam_t1_get_cont_tim](#) (void)
- void [p2_fw_sam_t1_set_rec_extraGuardTime](#) (uint8_t guardTime)
- void [p2_fw_sam_t1_set_bwi_cwi](#) (uint8_t bwi, uint8_t cwi)
- Bool [p2_fw_sam_t1_prepare_pps](#) (uint8_t *ppsData, uint8_t *ppsLen)
- void [p2_fw_sam_t1_init](#) (uint8_t chip, uint8_t mode)
- void [p2_fw_sam_t1_deinit](#) (void)
- void [p2_fw_sam_t1_start](#) (void)
- void [p2_fw_sam_t1_send](#) (uint8_t *data, uint32_t len, uint8_t frame_type)
- void [p2_fw_sam_t1_pps](#) (void)
- void [p2_fw_sam_t1_set_etu](#) (uint8_t fi_di)
- uint32_t [p2_fw_sam_t1_receive](#) (uint8_t *data, uint8_t frame_type)
- void [p2_fw_sam_t1_get_atr](#) (uint8_t *buffer, uint8_t *max_length)
- void [p2_fw_sam_t1_warm_reset](#) (void)
- Bool [p2_fw_sam_t1_is_busy](#) (void)
- Bool [p2_fw_sam_t1_is_power_off](#) (void)
- Bool [p2_fw_sam_t1_is_sam_inserted](#) (void)
- void [EINT3_IRQHandler](#) (void)
- void [TIMER2_IRQHandler](#) (void)
- void [TIMER0_IRQHandler](#) (void)

6.33.1 Define Documentation

6.33.1.1 `#define BITBAND_PERI(a, b) ((BITBAND_PERI_BASE + ((a)-BITBAND_PERI_REF)*32 + ((b)*4)))`

Definition at line 79 of file p3fw_samt1.c.

6.33.1.2 `#define BITBAND_PERI_BASE 0x42000000`

Definition at line 78 of file p3fw_samt1.c.

6.33.1.3 `#define BITBAND_PERI_REF 0x40000000`

Definition at line 77 of file p3fw_samt1.c.

6.33.1.4 `#define P2_FW_SAM_CLOCK_LEN_NS 222`

Definition at line 50 of file p3fw_samt1.c.

6.33.1.5 #define P2_FW_SAM_CONVERSION_DIRECT 1

Definition at line 90 of file p3fw_samt1.c.

6.33.1.6 #define P2_FW_SAM_CONVERSION_INVERSE 2

Definition at line 91 of file p3fw_samt1.c.

6.33.1.7 #define P2_FW_SAM_CONVERSION_UNKNOWN 0

Definition at line 89 of file p3fw_samt1.c.

6.33.1.8 #define P2_FW_SAM_DEBUG 0

Definition at line 37 of file p3fw_samt1.c.

6.33.1.9 #define P2_FW_SAM_DEFAULT_BWI 8405405

Definition at line 96 of file p3fw_samt1.c.

6.33.1.10 #define P2_FW_SAM_MAX_ATR_SIZE 33

Definition at line 93 of file p3fw_samt1.c.

6.33.1.11 #define P2_FW_SAM_MAX_REC_DATALEN 600

Definition at line 94 of file p3fw_samt1.c.

6.33.1.12 #define P2_FW_SAM_MODE_IDLE 4

Definition at line 73 of file p3fw_samt1.c.

6.33.1.13 #define P2_FW_SAM_MODE_POWER_OFF 7

Definition at line 75 of file p3fw_samt1.c.

6.33.1.14 #define P2_FW_SAM_MODE_RECEIVE 0

Definition at line 70 of file p3fw_samt1.c.

6.33.1.15 #define P2_FW_SAM_MODE_RESET 2

Definition at line 72 of file p3fw_samt1.c.

6.33.1.16 #define P2_FW_SAM_MODE_SEND 1

Definition at line 71 of file p3fw_samt1.c.

6.33.1.17 #define P2_FW_SAM_MODE_WAIT_START_BIT 6

Definition at line 74 of file p3fw_samt1.c.

6.33.1.18 #define P2_FW_SAM_PWM_CH0_MATCH_VALUE 4

Definition at line 47 of file p3fw_samt1.c.

6.33.1.19 #define P2_FW_SAM_PWM_CH1_MATCH_VALUE 2

Definition at line 48 of file p3fw_samt1.c.

6.33.1.20 #define P2_FW_SAM_T1_CLOCK_CONFIG 1

Definition at line 38 of file p3fw_samt1.c.

6.33.1.21 #define P2_FW_SAM_TIMER_PRESCALE_VALUE 4

Definition at line 49 of file p3fw_samt1.c.

6.33.1.22 #define T1_BYTE_WAIT_TIME 5

Definition at line 40 of file p3fw_samt1.c.

6.33.1.23 #define TIMER0_IR 0x40004000

Definition at line 81 of file p3fw_samt1.c.

6.33.1.24 #define TIMER0_IR_MR0 *((volatile unsigned char *) (BITBAND_PERI(TIMER0_IR,0)))

Definition at line 84 of file p3fw_samt1.c.

6.33.1.25 #define TIMER0_TCR 0x40004004

Definition at line 82 of file p3fw_samt1.c.

6.33.1.26 `#define TIMER0_TCR_ENABLE *((volatile unsigned char *)
(BITBAND_PERI(TIMER0_TCR,0)))`

Definition at line 85 of file p3fw_samt1.c.

6.33.1.27 `#define TIMER0_TCR_RESET *((volatile unsigned char *)
(BITBAND_PERI(TIMER0_TCR,1)))`

Definition at line 86 of file p3fw_samt1.c.

6.33.2 Function Documentation

6.33.2.1 `void EINT3_IRQHandler (void)`

Definition at line 815 of file p3fw_samt1.c.

6.33.2.2 `void p2_fw_sam_t1_deinit (void)`

Definition at line 382 of file p3fw_samt1.c.

6.33.2.3 `void p2_fw_sam_t1_get_atr (uint8_t * buffer, uint8_t * max_length)`

Definition at line 717 of file p3fw_samt1.c.

6.33.2.4 `uint32_t p2_fw_sam_t1_get_cont_tim (void)`

Definition at line 156 of file p3fw_samt1.c.

6.33.2.5 `uint8_t p2_fw_sam_t1_get_timing_mode (void)`

Definition at line 151 of file p3fw_samt1.c.

6.33.2.6 `void p2_fw_sam_t1_init (uint8_t chip, uint8_t mode)`

Definition at line 213 of file p3fw_samt1.c.

6.33.2.7 `Bool p2_fw_sam_t1_is_busy (void)`

Definition at line 778 of file p3fw_samt1.c.

6.33.2.8 `Bool p2_fw_sam_t1_is_power_off (void)`

Definition at line 798 of file p3fw_samt1.c.

6.33.2.9 Bool p2_fw_sam_t1_is_sam_inserted (void)

Definition at line 808 of file p3fw_samt1.c.

6.33.2.10 void p2_fw_sam_t1_pps (void)

Definition at line 608 of file p3fw_samt1.c.

6.33.2.11 Bool p2_fw_sam_t1_prepare_pps (uint8_t * ppsData, uint8_t * ppsLen)

Definition at line 178 of file p3fw_samt1.c.

6.33.2.12 uint32_t p2_fw_sam_t1_receive (uint8_t * data, uint8_t frame_type)

Definition at line 664 of file p3fw_samt1.c.

6.33.2.13 void p2_fw_sam_t1_send (uint8_t * data, uint32_t len, uint8_t frame_type)

Definition at line 486 of file p3fw_samt1.c.

6.33.2.14 void p2_fw_sam_t1_set_bwi_cwi (uint8_t bwi, uint8_t cwi)

Definition at line 174 of file p3fw_samt1.c.

6.33.2.15 void p2_fw_sam_t1_set_etu (uint8_t fi_di)

Definition at line 624 of file p3fw_samt1.c.

6.33.2.16 void p2_fw_sam_t1_set_my_debug (uint32_t x)

Definition at line 140 of file p3fw_samt1.c.

6.33.2.17 void p2_fw_sam_t1_set_rec_extraGuardTime (uint8_t guardTime)

Definition at line 161 of file p3fw_samt1.c.

6.33.2.18 void p2_fw_sam_t1_set_timing_mode (uint8_t mode)

Definition at line 145 of file p3fw_samt1.c.

6.33.2.19 void p2_fw_sam_t1_start (void)

Definition at line 422 of file p3fw_samt1.c.

6.33.2.20 void p2_fw_samt1_warm_reset (void)

Definition at line 738 of file p3fw_samt1.c.

6.33.2.21 void TIMER0_IRQHandler (void)

Definition at line 902 of file p3fw_samt1.c.

6.33.2.22 void TIMER2_IRQHandler (void)

Definition at line 852 of file p3fw_samt1.c.

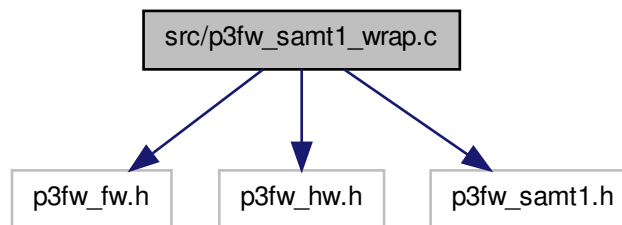
6.34 src/p3fw_samt1_wrap.c File Reference

```
#include <p3fw_fw.h>
```

```
#include <p3fw_hw.h>
```

```
#include <p3fw_samt1.h>
```

Include dependency graph for p3fw_samt1_wrap.c:

**Functions**

- phStatus_t [p3fw_samt1_init](#) (enum [p3fw_ic](#) ic, enum [p3fw_sam](#) sam)
- phStatus_t [p3fw_samt1_power_on](#) ()
- void [p3fw_samt1_set_param](#) (uint8_t FI_DI, uint8_t GuardTime, uint8_t BWI_CWI, uint8_t ClockStop, uint8_t IFSC)
- phStatus_t [p3fw_samt1_exchange](#) (uint8_t *in, uint32_t in_len, uint8_t *out, uint32_t *out_len)

6.34.1 Function Documentation

6.34.1.1 `phStatus_t p3fw_samt1_exchange (uint8_t * in, uint32_t in_len, uint8_t * out, uint32_t * out_len)`

Definition at line 99 of file p3fw_samt1_wrap.c.

6.34.1.2 `phStatus_t p3fw_samt1_init (enum p3fw_ic ic, enum p3fw_sam sam)`

Definition at line 23 of file p3fw_samt1_wrap.c.

6.34.1.3 `phStatus_t p3fw_samt1_power_on ()`

Definition at line 46 of file p3fw_samt1_wrap.c.

6.34.1.4 `void p3fw_samt1_set_param (uint8_t FI_DI, uint8_t GuardTime, uint8_t BWI_CWI, uint8_t ClockStop, uint8_t IFSC)`

Definition at line 75 of file p3fw_samt1_wrap.c.

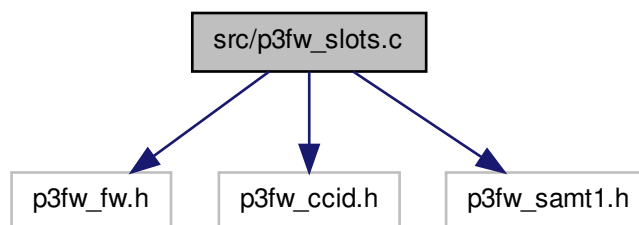
6.35 src/p3fw_slots.c File Reference

```
#include <p3fw_fw.h>
```

```
#include <p3fw_ccid.h>
```

```
#include <p3fw_samt1.h>
```

Include dependency graph for p3fw_slots.c:



Defines

- `#define P3FW_TEMP_BUFF_LEN 32`

Functions

- `phStatus_t p3fw_slots_init (void)`
Initializes the basic BLF structures and slots data.
- `void p3fw_slot_add_l3a_card (uint8_t bCLSlotIndex, uint8_t bSak)`
Adds a ISO14443 L3A card to the slot /c bCLSlotIndex.
- `void p3fw_slot_add_l3b_card (uint8_t bCLSlotIndex)`
Adds a ISO14443 L3B card to the slot /c bCLSlotIndex.
- `void p3fw_slot_add_l4_card (uint8_t bCLSlotIndex, uint8_t bSak)`
Adds a ISO14443 L4 card to the slot /c bCLSlotIndex.
- `void p3fw_slot_remove_cl_card (uint8_t bCLSlotIndex)`
Removes card from slot /c bCLSlotIndex.
- `phStatus_t p3fw_slot_get_atr (uint8_t bCCIDSlotIndex, uint8_t *pbAtrBuffer, uint8_t *pbMaxLength)`
Returns ATR for a slot /c bCCIDSlotIndex.
- `void p3fw_slot_reset_all_slots (uint8_t contact_card)`
Resets all slots to init state.

Variables

- `struct p3fw_hal_buffer aHalBuffers [P3FW_MAX_CONTACTLESS_SLOTS]`

6.35.1 Define Documentation

6.35.1.1 `#define P3FW_TEMP_BUFF_LEN 32`

Definition at line 23 of file p3fw_slots.c.

6.35.2 Variable Documentation

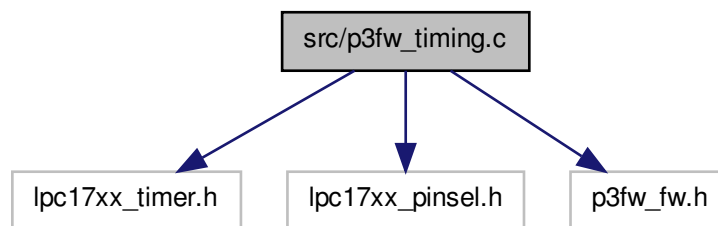
6.35.2.1 `struct p3fw_hal_buffer aHalBuffers[P3FW_MAX_CONTACTLESS_SLOTS]`

Definition at line 25 of file p3fw_slots.c.

6.36 src/p3fw_timing.c File Reference

```
#include <lpc17xx_timer.h>
#include <lpc17xx_pinsel.h>
#include <p3fw_fw.h>
```

Include dependency graph for p3fw_timing.c:



Functions

- void `p3fw_timing_init` (void)
Init's the timing interface.
- void `p3fw_timing_start` ()
Resets and starts the timer.
- uint32_t `p3fw_timing_stop` ()
Stops the timer and return the elapsed value.

Index

- [_DBC](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBD](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBD16](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBD32](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBG](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBGHL_](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBGH_](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBGL](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBGL_](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBG_](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBH](#)
 - [debug_frmwrk.h, 82](#)
 - [_DBH16](#)
 - [debug_frmwrk.h, 83](#)
 - [_DBH16L](#)
 - [debug_frmwrk.h, 83](#)
 - [_DBH32](#)
 - [debug_frmwrk.h, 83](#)
 - [_DBH32L](#)
 - [debug_frmwrk.h, 83](#)
 - [_DEBUG_FRMWRK_](#)
 - [debug_frmwrk.c, 109](#)
 - [_DG](#)
 - [debug_frmwrk.h, 83](#)
 - [_db_char](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
 - [_db_dec](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
 - [_db_dec_16](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
- [_db_dec_32](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
- [_db_get_char](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
- [_db_hex](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
- [_db_hex_16](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
- [_db_hex_32](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
- [_db_mhex](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
- [_db_msg](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 87](#)
- [_db_msg_](#)
 - [debug_frmwrk.c, 113](#)
 - [debug_frmwrk.h, 88](#)
- [_p3fw_cc_slot, 59](#)
 - [bPPP, 59](#)
 - [eProtocolType, 59](#)
 - [eSlotType, 59](#)
- [_p3fw_cl_slot, 60](#)
 - [bAts, 60](#)
 - [bBlockPoll, 60](#)
 - [bPPP, 61](#)
 - [bSak, 61](#)
 - [bUid, 61](#)
 - [bUidLength, 61](#)
 - [eProtocolType, 61](#)
 - [eSlotType, 61](#)
 - [pHal, 61](#)
 - [sISO14443L3a, 61](#)

- sISO14443L3b, 62
- sISO14443L4, 62
- sISO14443L4a, 62
- sMfc, 62
- sMful, 62
- sMifare, 62
- aBuffer
 - p3fw_data, 71
- aHalBuffers
 - p3fw_dm_hal.c, 125
 - p3fw_slots.c, 147
- apdu
 - P3FW_APDU_CC_EXT_INS, 8
 - P3FW_APDU_CC_EXT_MANAGE_SESSION, 8
 - P3FW_APDU_CC_EXT_SWITCH_PROTOCOL, 8
 - P3FW_APDU_CC_EXT_TRANS_EXCHANGE, 8
 - P3FW_APDU_CLASS, 8
 - P3FW_APDU_G_AUTH_CMD_INS, 8
 - P3FW_APDU_GET_DATA_INS, 8
 - P3FW_APDU_INS, 8
 - P3FW_APDU_Lc, 8
 - P3FW_APDU_Le, 8
 - P3FW_APDU_LOAD_KEY_INS, 9
 - P3FW_APDU_P1, 9
 - P3FW_APDU_P2, 9
 - P3FW_APDU_PAYLOAD, 9
 - P3FW_APDU_READ_BIN, 9
 - P3FW_APDU_UPDATE_BIN, 9
- APDUs Defines, 7
- bAts
 - _p3fw_cl_slot, 60
- bAuth
 - p3fw_data, 71
- bBlockPoll
 - _p3fw_cl_slot, 60
- bCLLastNotifyEvent
 - p3fw_data, 71
- bConfig
 - p3fw_data, 71
- bDoPoll
 - p3fw_data, 71
- bfl
 - p3fw_data, 71
- bh
 - p3_fw_ccid_exec, 67
- bIsBusy
 - p3fw_data, 72
- BITBAND_PERI
 - p3fw_samt1.c, 140
- BITBAND_PERI_BASE
 - p3fw_samt1.c, 140
- BITBAND_PERI_REF
 - p3fw_samt1.c, 140
- bitIndex
 - p2_fw_SAM_ctrl_, 65
- bKeyType
 - p3fw_data, 72
- bMaxSlots
 - p3fw_data, 72
- bMFCKey
 - p3fw_data, 72
- bMFCKeyLength
 - p3fw_data, 72
- bPPP
 - _p3fw_cc_slot, 59
 - _p3fw_cl_slot, 61
- bSak
 - _p3fw_cl_slot, 61
- bSlot
 - p3_fw_ccid_exec, 67
- bUid
 - _p3fw_cl_slot, 61
- bUidLength
 - _p3fw_cl_slot, 61
- bwi
 - p2_fw_SAM_ctrl_, 65
- BWI_CWI
 - samt1_param, 77
- cc
 - p3fw_data, 72
- ccActiveSlotNum
 - p3fw_data, 72
- ccid
 - p3fw_ccid_bh_escape, 15
 - p3fw_ccid_bh_get_parameters, 16
 - p3fw_ccid_bh_get_slot_status, 16
 - p3fw_ccid_bh_icc_power_off, 16
 - p3fw_ccid_bh_icc_power_on, 16
 - p3fw_ccid_bh_set_parameters, 17
 - p3fw_ccid_bh_xfer, 17
 - P3FW_CCID_BULK_HEADER, 11
 - P3FW_CCID_BULK_OUT_REQ_ESCAPE, 11

- P3FW_CCID_BULK_OUT_REQ_GETPARAMETERS, [17](#)
- [12](#) P3FW_CCID_INT_IN_NOTIFY_SLOT_-
- P3FW_CCID_BULK_OUT_REQ_GETSLOTSTATUS, [15](#)
- [12](#) P3FW_CCID_RDR_TO_PC_ESCAPE,
- P3FW_CCID_BULK_OUT_REQ_ICCPOWERON, [15](#)
- [12](#) P3FW_CCID_RDR_TO_PC_PARAMETERS,
- P3FW_CCID_BULK_OUT_REQ_ICCPOWERON, [15](#)
- [12](#) P3FW_CCID_RDR_TO_PC_SLOT_-
- P3FW_CCID_BULK_OUT_REQ_SETPARAMETERS_BLOCK, [15](#)
- [12](#) P3FW_CCID_RDR_TO_PC_SLOT_-
- P3FW_CCID_BULK_OUT_REQ_XFRBLOCK_STATUS, [15](#)
- [12](#) p3fw_ccid_send_frame, [18](#)
- P3FW_CCID_CARD_IN_SLOT, [12](#) p3fw_ccid_send_notify, [18](#)
- P3FW_CCID_CARD_OUT_SLOT, [12](#) p3fw_ccid_set_output_payload_length,
- P3FW_CCID_ERROR_SLOT_BUSY, [18](#)
- [13](#) P3FW_CCID_STATUS_CMD_FAILED,
- P3FW_CCID_ERROR_SLOT_CMD_- [15](#)
- NOT_SUPPORTED, [13](#) p3fw_ccid_th_dispatch, [18](#)
- P3FW_CCID_ERROR_SLOT_HW_- CCID functions, [9](#)
- ERROR, [13](#) chipMode
- P3FW_CCID_ERROR_SLOT_ICC_- p2_fw_SAM_ctrl_, [65](#)
- MUTE, [13](#) chipType
- P3FW_CCID_ERROR_SLOT_NOT_- p2_fw_SAM_ctrl_, [65](#)
- EXIST, [13](#) cl
- P3FW_CCID_ERROR_SLOT_XFR_- p3fw_data, [72](#)
- OVERRUN, [13](#) clActiveSlotNum
- p3fw_ccid_get_input_payload_buffer, p3fw_data, [72](#)
- [17](#) ClockStop
- p3fw_ccid_get_input_payload_length, sam_t1_param, [77](#)
- [17](#) comm
- p3fw_ccid_get_output_payload_buffer, p3fw_data, [72](#)
- [17](#) config
- P3FW_CCID_HEADER_LENGTH_- p3fw_data, [73](#)
- BYTE_1, [13](#) cont_tim
- P3FW_CCID_HEADER_LENGTH_- p2_fw_SAM_ctrl_, [65](#)
- BYTE_2, [13](#) conversion
- P3FW_CCID_HEADER_LENGTH_- p2_fw_SAM_ctrl_, [65](#)
- BYTE_3, [14](#) cpot_atr_frame, [63](#)
- P3FW_CCID_HEADER_LENGTH_- historicalC, [63](#)
- BYTE_4, [14](#) state, [63](#)
- P3FW_CCID_HEADER_MESSAGE_- TAI, [63](#)
- TYPE, [14](#) TBI, [63](#)
- P3FW_CCID_HEADER_MSG_BYTE_- TCi, [63](#)
- 1, [14](#) TCK, [63](#)
- P3FW_CCID_HEADER_MSG_BYTE_- TDi, [63](#)
- 2, [14](#)
- P3FW_CCID_HEADER_MSG_BYTEdata
- 3, [14](#) p2_fw_SAM_ctrl_, [65](#)
- P3FW_CCID_HEADER_SEQ, [14](#) dataIndex
- P3FW_CCID_HEADER_SLOT, [14](#) p2_fw_SAM_ctrl_, [65](#)

- debug_frmwrk.c
 - _DEBUG_FRMWRK_, 109
 - _db_char, 113
 - _db_dec, 113
 - _db_dec_16, 113
 - _db_dec_32, 113
 - _db_get_char, 113
 - _db_hex, 113
 - _db_hex_16, 113
 - _db_hex_32, 113
 - _db_mhex, 113
 - _db_msg, 113
 - _db_msg_, 113
 - debug_frmwrk_init, 109
 - UARTGetChar, 109
 - UARTPutChar, 110
 - UARTPutDec, 110
 - UARTPutDec16, 110
 - UARTPutDec32, 110
 - UARTPutHex, 111
 - UARTPutHex16, 111
 - UARTPutHex32, 111
 - UARTPutHexMulti, 112
 - UARTPuts, 112
 - UARTPuts_, 112
- debug_frmwrk.h
 - _DBC, 82
 - _DBD, 82
 - _DBD16, 82
 - _DBD32, 82
 - _DBG, 82
 - _DBGHL_, 82
 - _DBGH_, 82
 - _DBGL, 82
 - _DBGL_, 82
 - _DBG_, 82
 - _DBH, 82
 - _DBH16, 83
 - _DBH16L, 83
 - _DBH32, 83
 - _DBH32L, 83
 - _DG, 83
 - _db_char, 87
 - _db_dec, 87
 - _db_dec_16, 87
 - _db_dec_32, 87
 - _db_get_char, 87
 - _db_hex, 87
 - _db_hex_16, 87
 - _db_hex_32, 87
 - _db_mhex, 87
 - _db_msg, 87
 - _db_msg_, 88
 - debug_frmwrk_init, 83
 - DEBUG_UART_PORT, 83
 - UARTGetChar, 83
 - UARTPutChar, 84
 - UARTPutDec, 84
 - UARTPutDec16, 84
 - UARTPutDec32, 85
 - UARTPutHex, 85
 - UARTPutHex16, 85
 - UARTPutHex32, 86
 - UARTPutHexMulti, 86
 - UARTPuts, 86
 - UARTPuts_, 86
- debug_frmwrk_init
 - debug_frmwrk.c, 109
 - debug_frmwrk.h, 83
- DEBUG_UART_PORT
 - debug_frmwrk.h, 83
- Direct
 - P3FW_DM_CHECK_LENGTH_EQUAL, 22
 - P3FW_DM_CHECK_LENGTH_LARGER, 22
 - P3FW_DM_CHECK_STATUS, 22
 - P3FW_DM_CID, 22
 - p3fw_dm_cid, 33
 - P3FW_DM_CID_FREE, 23
 - P3FW_DM_CID_GET_FREE, 23
 - P3FW_DM_CID_INIT, 23
 - P3FW_DM_CONTACT_CARD, 23
 - p3fw_dm_contact_card, 33
 - P3FW_DM_CONTACTCARD_ACTIVATE_CARD, 23
 - P3FW_DM_CONTACTCARD_CLOCK_START, 23
 - P3FW_DM_CONTACTCARD_CLOCK_STOP, 23
 - P3FW_DM_CONTACTCARD_COLD_RESET, 23
 - P3FW_DM_CONTACTCARD_DEACTIVATE_CARD, 24
 - P3FW_DM_CONTACTCARD_PPS, 24
 - P3FW_DM_CONTACTCARD_PRESENCE_CHECK, 24
 - P3FW_DM_CONTACTCARD_TRANSMIT_DATA, 24

- P3FW_DM_CONTACTCARD_WARM_RESET, [24](#)
- p3fw_dm_get_class, [34](#)
- p3fw_dm_get_input_payload, [34](#)
- p3fw_dm_get_input_payload_length, [34](#)
- p3fw_dm_get_input_pointer_uint8_t, [34](#)
- p3fw_dm_get_input_uint16_t, [34](#)
- p3fw_dm_get_input_uint32_t, [34](#)
- p3fw_dm_get_input_uint8_t, [34](#)
- p3fw_dm_get_instruction, [35](#)
- p3fw_dm_get_output_payload, [35](#)
- p3fw_dm_get_output_pointer_uint8_t, [35](#)
- p3fw_dm_get_slot_index, [35](#)
- P3FW_DM_HAL, [24](#)
- p3fw_dm_hal, [35](#)
- P3FW_DM_HAL_APP_PROT_SET, [24](#)
- P3FW_DM_HAL_EXEC_CMD, [24](#)
- P3FW_DM_HAL_GET_CFG, [25](#)
- P3FW_DM_HAL_INIT, [25](#)
- P3FW_DM_HAL_MFC_AUTH, [25](#)
- P3FW_DM_HAL_MFC_AUTH_KEY, [25](#)
- P3FW_DM_HAL_SET_CFG, [25](#)
- P3FW_DM_HAL_WAIT, [25](#)
- P3FW_DM_HAL_XCHG, [25](#)
- p3fw_dm_keystore, [35](#)
- P3FW_DM_KSTOR, [35](#)
- P3FW_DM_KSTOR_CHG_KUC, [26](#)
- P3FW_DM_KSTOR_FORMAT_KEY, [26](#)
- P3FW_DM_KSTOR_GET_CFG_STR, [26](#)
- P3FW_DM_KSTOR_GET_CONFIG, [26](#)
- P3FW_DM_KSTOR_GET_KEY, [26](#)
- P3FW_DM_KSTOR_GET_KEY_ENTRY, [26](#)
- P3FW_DM_KSTOR_GET_KUC, [26](#)
- P3FW_DM_KSTOR_INIT, [26](#)
- P3FW_DM_KSTOR_SET_CFG_STR, [27](#)
- P3FW_DM_KSTOR_SET_CONFIG, [27](#)
- P3FW_DM_KSTOR_SET_FULL_KEY, [27](#)
- P3FW_DM_KSTOR_SET_KEY, [27](#)
- P3FW_DM_KSTOR_SET_KEY_POS, [27](#)
- P3FW_DM_KSTOR_SET_KUC, [27](#)
- P3FW_DM_L3, [27](#)
- p3fw_dm_l3, [35](#)
- P3FW_DM_L3_ACT_CARD, [27](#)
- P3FW_DM_L3_ANTICOL, [28](#)
- P3FW_DM_L3_GET_SER, [28](#)
- P3FW_DM_L3_HLTA, [28](#)
- P3FW_DM_L3_INIT, [28](#)
- P3FW_DM_L3_REQA, [28](#)
- P3FW_DM_L3_SELECT, [28](#)
- P3FW_DM_L3_WKUA, [28](#)
- P3FW_DM_L3_XCHG, [28](#)
- P3FW_DM_L4, [29](#)
- p3fw_dm_l4, [36](#)
- P3FW_DM_L4_DESELECT, [29](#)
- P3FW_DM_L4_GET_CFG, [29](#)
- P3FW_DM_L4_INIT, [29](#)
- P3FW_DM_L4_PRESET_CHECK, [29](#)
- P3FW_DM_L4_RESET_PROTO, [29](#)
- P3FW_DM_L4_SET_CFG, [29](#)
- P3FW_DM_L4_SET_PROTO, [29](#)
- P3FW_DM_L4_XCHG, [30](#)
- P3FW_DM_L4A, [30](#)
- p3fw_dm_l4a, [36](#)
- P3FW_DM_L4A_ACT_CARD, [30](#)
- P3FW_DM_L4A_GET_PROTO_PARM, [30](#)
- P3FW_DM_L4A_INIT, [30](#)
- P3FW_DM_L4A_PPS, [30](#)
- P3FW_DM_L4A_RATS, [30](#)
- P3FW_DM_RO, [30](#)
- p3fw_dm_ro, [36](#)
- P3FW_DM_RO_CONF_OVER, [31](#)
- P3FW_DM_RO_FIELD_OFF, [31](#)
- P3FW_DM_RO_FIELD_ON, [31](#)
- P3FW_DM_RO_FIELD_RESET, [31](#)
- P3FW_DM_RO_GET_CONF, [31](#)
- P3FW_DM_RO_GET_STATUS, [31](#)
- P3FW_DM_RO_LEDS_OFF, [31](#)
- P3FW_DM_RO_LEDS_ON, [31](#)
- P3FW_DM_RO_READ_REG, [32](#)
- P3FW_DM_RO_RESET, [32](#)
- P3FW_DM_RO_SET_CONF, [32](#)
- P3FW_DM_RO_SET_PCSC_MODE, [32](#)
- P3FW_DM_RO_TEST_MODE, [32](#)
- P3FW_DM_RO_WRITE_REG, [32](#)
- p3fw_dm_send_frame, [36](#)

- P3FW_DM_XCHG, 32
- p3fw_dm_xchg, 36
- P3FW_DM_XCHG_INIT, 32
- P3FW_DM_XCHG_L3, 33
- P3FW_DM_XCHG_L4, 33
- P3FW_DM_XCHG_MFC_AUTH, 33
- P3FW_DM_XCHG_MFC_AUTH_KEY, 33
- P3FW_DM_XCHG_PC, 33
- P3FW_DM_XCHG_RAW, 33
- p3fw_dm_xfer, 36
- dwIndex
 - p3fw_data, 73
- dwLength
 - p3fw_data, 73
- dwWrittenLength
 - p3fw_data, 73
- eBalCfg
 - p3fw_data, 73
- eClSlotsType
 - p3fw_data, 73
- eExtIf
 - p3fw_data, 73
- EINT3_IRQHandler
 - p3fw_samt1.c, 143
 - p3fw_samt1.h, 104
- eMode
 - p3fw_data, 73
- ePcScMode
 - p3fw_data, 73
- eProtocolType
 - _p3fw_cc_slot, 59
 - _p3fw_cl_slot, 61
- eReaderIC
 - p3fw_data, 73
- eSam
 - p3fw_data, 74
- eSlotType
 - _p3fw_cc_slot, 59
 - _p3fw_cl_slot, 61
- etu
 - p2_fw_SAM_ctrl_, 65
- FI_DI
 - sam_t1_param, 77
- Firmware, 37
- GuardTime
 - sam_t1_param, 77
- Hardware
 - p3fw_hw_ctrl_signal, 54
 - p3fw_hw_init, 54
 - P3FW_HW_SIGNAL_ANTENNA, 54
 - P3FW_HW_SIGNAL_BEEPER, 54
 - P3FW_HW_SIGNAL_YELLOW_2, 54
 - P3FW_HW_SIGNAL_YELLOW_3, 54
 - P3FW_HW_SIGNAL_YELLOW_4, 54
- historicalC
 - cpot_atr_frame, 63
- IAP
 - p3fw_flash.c, 136
- IAP_BLANK_CHECK_SECTOR
 - p3fw_flash.c, 134
- IAP_COMPARE
 - p3fw_flash.c, 134
- IAP_COPY_RAM_TO_FLASH
 - p3fw_flash.c, 134
- IAP_ERASE_SECTOR
 - p3fw_flash.c, 134
- IAP_LOCATION
 - p3fw_flash.c, 134
- IAP_PREP_SECTORS
 - p3fw_flash.c, 134
- IAP_READ_BOOT_CODE_VER
 - p3fw_flash.c, 134
- IAP_READ_DEV_SER_NUM
 - p3fw_flash.c, 134
- IAP_READ_ID
 - p3fw_flash.c, 134
- IAP_REINVOKE
 - p3fw_flash.c, 135
- IAP_SEC_29
 - p3fw_flash.c, 135
- IFSC
 - sam_t1_param, 77
- in
 - p3fw_data, 74
- include/debug_frmwrk.h, 79
- include/p3fw_apdu.h, 88
- include/p3fw_ccid.h, 89
- include/p3fw_dm.h, 91
- include/p3fw_fw.h, 95
- include/p3fw_hw.h, 99
- include/p3fw_pcsc.h, 100
- include/p3fw_pins.h, 101

- include/p3fw_samt1.h, 101
- Initialization, 53
- ipc
 - p3fw_data, 74
- len
 - p2_fw_SAM_ctrl_, 65
- main
 - p3fw.c, 116
- mode
 - p2_fw_SAM_ctrl_, 66
- Mode Functions, 19
- out
 - p3fw_data, 74
- P2_FW_SAM_CLOCK_LEN_NS
 - p3fw_samt1.c, 140
- P2_FW_SAM_CONVERSION_DIRECT
 - p3fw_samt1.c, 140
- P2_FW_SAM_CONVERSION_INVERSE
 - p3fw_samt1.c, 141
- P2_FW_SAM_CONVERSION_UNKNOWN
 - p3fw_samt1.c, 141
- p2_fw_SAM_ctrl_, 64
 - bitIndex, 65
 - bwi, 65
 - chipMode, 65
 - chipType, 65
 - cont_tim, 65
 - conversion, 65
 - data, 65
 - dataIndex, 65
 - etu, 65
 - len, 65
 - mode, 66
 - parityCount, 66
 - recExtraGuardTime, 66
 - send_data, 66
 - sendLen, 66
 - sendSeqData, 66
 - SessionATR, 66
 - SessionATR_Size, 66
 - t0, 66
 - timingMode, 66
 - tmpByteWait_time, 66
- P2_FW_SAM_DEBUG
 - p3fw_samt1.c, 141
- P2_FW_SAM_DEFAULT_BWI
 - p3fw_samt1.c, 141
- P2_FW_SAM_FRAME_APDU
 - p3fw_samt1.h, 102
- P2_FW_SAM_FRAME_T1
 - p3fw_samt1.h, 102
- P2_FW_SAM_MAX_ATR_SIZE
 - p3fw_samt1.c, 141
- P2_FW_SAM_MAX_REC_DATALEN
 - p3fw_samt1.c, 141
- P2_FW_SAM_MODE_IDLE
 - p3fw_samt1.c, 141
- P2_FW_SAM_MODE_POWER_OFF
 - p3fw_samt1.c, 141
- P2_FW_SAM_MODE_PPS
 - p3fw_samt1.h, 102
- P2_FW_SAM_MODE_RECEIVE
 - p3fw_samt1.c, 141
- P2_FW_SAM_MODE_RESET
 - p3fw_samt1.c, 141
- P2_FW_SAM_MODE_SEND
 - p3fw_samt1.c, 141
- P2_FW_SAM_MODE_WAIT_START_BIT
 - p3fw_samt1.c, 142
- P2_FW_SAM_MODE_X
 - p3fw_samt1.h, 102
- P2_FW_SAM_PWM_CH0_MATCH_VALUE
 - p3fw_samt1.c, 142
- P2_FW_SAM_PWM_CH1_MATCH_VALUE
 - p3fw_samt1.c, 142
- P2_FW_SAM_T1_CLOCK_CONFIG
 - p3fw_samt1.c, 142
- p2_fw_sam_t1_deinit
 - p3fw_samt1.c, 143
 - p3fw_samt1.h, 104
- p2_fw_sam_t1_get_atr
 - p3fw_samt1.c, 143
 - p3fw_samt1.h, 104
- p2_fw_sam_t1_get_cont_tim
 - p3fw_samt1.c, 143
 - p3fw_samt1.h, 104
- p2_fw_sam_t1_get_timing_mode
 - p3fw_samt1.c, 143
 - p3fw_samt1.h, 104
- p2_fw_sam_t1_init
 - p3fw_samt1.c, 143
 - p3fw_samt1.h, 104
- p2_fw_sam_t1_is_busy
 - p3fw_samt1.c, 143
 - p3fw_samt1.h, 105
- p2_fw_sam_t1_is_power_off

- p3fw_samt1.c, [143](#)
- p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_is_sam_inserted
 - p3fw_samt1.c, [143](#)
 - p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_pps
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_prepare_pps
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_receive
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_send
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_set_bwi_cwi
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_set_etu
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_set_my_debug
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_set_rec_extraGuardTime
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [105](#)
- p2_fw_sam_t1_set_timing_mode
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [106](#)
- p2_fw_sam_t1_start
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [106](#)
- p2_fw_sam_t1_warm_reset
 - p3fw_samt1.c, [144](#)
 - p3fw_samt1.h, [106](#)
- P2_FW_SAM_TIMER_PRESCALE_VALUE
 - p3fw_samt1.c, [142](#)
- P2_FW_TIMING_MODE_COM
 - p3fw_samt1.h, [103](#)
- P2_FW_TIMING_MODE_FDT
 - p3fw_samt1.h, [103](#)
- P2_FW_TIMING_MODE_NONE
 - p3fw_samt1.h, [103](#)
- p3_fw_ccid_exec, [67](#)
 - bh, [67](#)
 - bSlot, [67](#)
- p3fw
 - Pegoda, [53](#)
- p3fw.c
 - main, [116](#)
 - P3FW_DIP_CFG_BAL_I2C, [115](#)
 - P3FW_DIP_CFG_BAL_SPI, [115](#)
 - P3FW_DIP_CFG_BAL_UART, [115](#)
 - P3FW_DIP_CFG_SAM_IN_X, [115](#)
 - P3FW_DIP_CFG_SAM_NO, [115](#)
 - P3FW_DIP_CFG_SAM_NON_X, [115](#)
 - P3FW_DIP_EXT_IF_ETHERNET, [115](#)
 - P3FW_DIP_EXT_IF_RS232, [115](#)
 - P3FW_DIP_EXT_IF_RS485, [115](#)
 - P3FW_DIP_EXT_IF_USB, [116](#)
 - P3FW_DIP_MODE_ACT_BOOTLOADER, [116](#)
 - P3FW_DIP_MODE_CFG_OVERWRITE, [116](#)
 - P3FW_DIP_MODE_DEMO, [116](#)
 - P3FW_DIP_MODE_PCSC, [116](#)
 - p3fw_dump_regs, [116](#)
 - P3FW_TASK_START_ERROR, [116](#)
 - P3FW_TASK_START_EXECUTE, [116](#)
 - P3FW_TASK_START_POLL, [116](#)
- P3FW_BAL_I2C
 - Pegoda, [46](#)
- P3FW_BAL_SPI
 - Pegoda, [46](#)
- P3FW_BAL_UART
 - Pegoda, [46](#)
- P3FW_EXT_IF_ETHERNET
 - Pegoda, [46](#)
- P3FW_EXT_IF_RS232
 - Pegoda, [46](#)
- P3FW_EXT_IF_RS485
 - Pegoda, [46](#)
- P3FW_EXT_IF_USB
 - Pegoda, [46](#)
- P3FW_IC_RC523
 - Pegoda, [46](#)
- P3FW_IC_RC663
 - Pegoda, [46](#)
- P3FW_IC_SAM
 - Pegoda, [46](#)
- P3FW_MODE_ACTIVATE_BOOTLOADER
 - Pegoda, [47](#)
- P3FW_MODE_CFG_OVERWRITE
 - Pegoda, [47](#)
- P3FW_MODE_DEMO
 - Pegoda, [46](#)
- P3FW_MODE_PCSC

- Pegoda, [46](#)
- P3FW_PCSC_MODE_DIRECT
 - Pegoda, [47](#)
- P3FW_PCSC_MODE_NORMAL
 - Pegoda, [47](#)
- P3FW_PCSC_PROTOCOL_RAW
 - Pegoda, [47](#)
- P3FW_PCSC_PROTOCOL_T0
 - Pegoda, [47](#)
- P3FW_PCSC_PROTOCOL_T1
 - Pegoda, [47](#)
- P3FW_PCSC_PROTOCOL_UNKNOWN
 - Pegoda, [47](#)
- P3FW_SAM_IN_X
 - Pegoda, [47](#)
- P3FW_SAM_NO
 - Pegoda, [47](#)
- P3FW_SAM_NON_X
 - Pegoda, [47](#)
- P3FW_SLOT_EMPTY
 - Pegoda, [47](#)
- P3FW_SLOT_ISO1444L3A_CARD
 - Pegoda, [48](#)
- P3FW_SLOT_ISO1444L3B_CARD
 - Pegoda, [48](#)
- P3FW_SLOT_ISO1444L4_CARD
 - Pegoda, [48](#)
- P3FW_SLOT_SAM
 - Pegoda, [47](#)
- P3FW_APDU_CC_EXT_INS
 - apdu, [8](#)
- P3FW_APDU_CC_EXT_MANAGE_SESSION
 - apdu, [8](#)
- P3FW_APDU_CC_EXT_SWITCH_PROTOCOL
 - apdu, [8](#)
- P3FW_APDU_CC_EXT_TRANS_EXCHANGE
 - apdu, [8](#)
- P3FW_APDU_CLASS
 - apdu, [8](#)
- P3FW_APDU_G_AUTH_CMD_INS
 - apdu, [8](#)
- P3FW_APDU_GET_DATA_INS
 - apdu, [8](#)
- P3FW_APDU_INS
 - apdu, [8](#)
- P3FW_APDU_Lc
 - apdu, [8](#)
- P3FW_APDU_Le
 - apdu, [8](#)
- P3FW_APDU_LOAD_KEY_INS
 - apdu, [9](#)
- P3FW_APDU_P1
 - apdu, [9](#)
- P3FW_APDU_P2
 - apdu, [9](#)
- P3FW_APDU_PAYLOAD
 - apdu, [9](#)
- P3FW_APDU_READ_BIN
 - apdu, [9](#)
- P3FW_APDU_UPDATE_BIN
 - apdu, [9](#)
- p3fw_bal_configuration
 - Pegoda, [46](#)
- P3FW_BUILD
 - Pegoda, [40](#)
- p3fw_cc_slot
 - Pegoda, [45](#)
- p3fw_ccid_bh_escape
 - ccid, [15](#)
- p3fw_ccid_bh_get_parameters
 - ccid, [16](#)
- p3fw_ccid_bh_get_slot_status
 - ccid, [16](#)
- p3fw_ccid_bh_icc_power_off
 - ccid, [16](#)
- p3fw_ccid_bh_icc_power_on
 - ccid, [16](#)
- p3fw_ccid_bh_set_parameters
 - ccid, [17](#)
- p3fw_ccid_bh_xfer
 - ccid, [17](#)
- P3FW_CCID_BULK_HEADER
 - ccid, [11](#)
- P3FW_CCID_BULK_OUT_REQ_ESCAPE
 - ccid, [11](#)
- P3FW_CCID_BULK_OUT_REQ_GETPARAMETERS
 - ccid, [12](#)
- P3FW_CCID_BULK_OUT_REQ_GETSLOTSTATUS
 - ccid, [12](#)
- P3FW_CCID_BULK_OUT_REQ_ICCPOWEROFF
 - ccid, [12](#)
- P3FW_CCID_BULK_OUT_REQ_ICCPOWERON
 - ccid, [12](#)
- P3FW_CCID_BULK_OUT_REQ_SETPARAMETERS
 - ccid, [12](#)
- P3FW_CCID_BULK_OUT_REQ_XFRBLOCK
 - ccid, [12](#)
- P3FW_CCID_CARD_IN_SLOT
 - ccid, [12](#)
- P3FW_CCID_CARD_OUT_SLOT

- ccid, [12](#)
- P3FW_CCID_ERROR_SLOT_BUSY ccid, [13](#)
- P3FW_CCID_ERROR_SLOT_CMD_NOT_SUPPORTED ccid, [13](#)
- P3FW_CCID_ERROR_SLOT_HW_ERROR ccid, [13](#)
- P3FW_CCID_ERROR_SLOT_ICC_MUTE ccid, [13](#)
- P3FW_CCID_ERROR_SLOT_NOT_EXIST ccid, [13](#)
- P3FW_CCID_ERROR_SLOT_XFR_OVERFLOW ccid, [13](#)
- p3fw_ccid_get_input_payload_buffer ccid, [17](#)
- p3fw_ccid_get_input_payload_length ccid, [17](#)
- p3fw_ccid_get_output_payload_buffer ccid, [17](#)
- P3FW_CCID_HEADER_LENGTH_BYTE_1 ccid, [13](#)
- P3FW_CCID_HEADER_LENGTH_BYTE_2 ccid, [13](#)
- P3FW_CCID_HEADER_LENGTH_BYTE_3 ccid, [14](#)
- P3FW_CCID_HEADER_LENGTH_BYTE_4 ccid, [14](#)
- P3FW_CCID_HEADER_MESSAGE_TYPE ccid, [14](#)
- P3FW_CCID_HEADER_MSG_BYTE_1 ccid, [14](#)
- P3FW_CCID_HEADER_MSG_BYTE_2 ccid, [14](#)
- P3FW_CCID_HEADER_MSG_BYTE_3 ccid, [14](#)
- P3FW_CCID_HEADER_SEQ ccid, [14](#)
- P3FW_CCID_HEADER_SLOT ccid, [14](#)
- p3fw_ccid_input_header_get_byte ccid, [17](#)
- P3FW_CCID_INT_IN_NOTIFY_SLOT_CHANGE ccid, [15](#)
- P3FW_CCID_RDR_TO_PC_ESCAPE ccid, [15](#)
- P3FW_CCID_RDR_TO_PC_PARAMETERS ccid, [15](#)
- P3FW_CCID_RDR_TO_PC_SLOT_DATA_BLOCK ccid, [15](#)
- P3FW_CCID_RDR_TO_PC_SLOT_STATUS ccid, [15](#)
- p3fw_ccid_send_frame ccid, [18](#)
- p3fw_ccid_send_notify ccid, [18](#)
- p3fw_ccid_set_output_payload_length ccid, [18](#)
- P3FW_CCID_STATUS_CMD_FAILED ccid, [15](#)
- p3fw_ccid_th_dispatch ccid, [18](#)
- P3FW_CFG_BOOTLOADER_ACTIVE Pegoda, [40](#)
- P3FW_CFG_BOOTLOADER_VERSION Pegoda, [40](#)
- P3FW_CFG_CONT_TIMING Pegoda, [41](#)
- P3FW_CFG_COUNT p3fw_flash.c, [135](#)
- P3FW_CFG_ENTRY_OFFSET p3fw_flash.c, [135](#)
- P3FW_CFG_ENTRY_SIZE p3fw_flash.c, [135](#)
- P3FW_CFG_GET_CONT_TIMING Pegoda, [41](#)
- P3FW_CFG_MAX p3fw_flash.c, [135](#)
- P3FW_CFG_MAX_SLOTS Pegoda, [41](#)
- P3FW_CFG_S1 p3fw_flash.c, [135](#)
- P3FW_CFG_S2 p3fw_flash.c, [135](#)
- P3FW_CFG_SET_DIP_SWITCHES Pegoda, [41](#)
- P3FW_CHECK_STATUS Pegoda, [41](#)
- p3fw_cl_slot Pegoda, [45](#)
- P3FW_CONTACT_SLOTS_MASK Pegoda, [41](#)
- p3fw_data, [67](#)
- aBuffer, [71](#)

- bAuth, [71](#)
- bCLLastNotifyEvent, [71](#)
- bConfig, [71](#)
- bDoPoll, [71](#)
- bfl, [71](#)
- bIsBusy, [72](#)
- bKeyType, [72](#)
- bMaxSlots, [72](#)
- bMFCKey, [72](#)
- bMFCKeyLength, [72](#)
- cc, [72](#)
- ccActiveSlotNum, [72](#)
- cl, [72](#)
- clActiveSlotNum, [72](#)
- comm, [72](#)
- config, [73](#)
- dwIndex, [73](#)
- dwLength, [73](#)
- dwWrittenLength, [73](#)
- eBalCfg, [73](#)
- eCISlotsType, [73](#)
- eExtIf, [73](#)
- eMode, [73](#)
- ePcScMode, [73](#)
- eReaderIC, [73](#)
- eSam, [74](#)
- in, [74](#)
- ipc, [74](#)
- out, [74](#)
- pBal, [74](#)
- pcsc, [74](#)
- sBalI2c, [74](#)
- sBalSerial, [74](#)
- sBalSpi, [74](#)
- sCidManager, [74](#)
- send, [74](#)
- sKeyEntry, [75](#)
- sKeyKUC, [75](#)
- sKeyStore, [75](#)
- sKeyVersion, [75](#)
- slots, [75](#)
- sMultexBal, [75](#)
- sQueueCCIDExec, [75](#)
- p3fw_demo.c
 - P3FW_DEMO_ATQB_LEN, [121](#)
 - P3FW_DEMO_RX_BUFFER, [121](#)
 - P3FW_DEMO_TX_BUFFER, [121](#)
 - P3FW_DEMO_UID_LEN, [121](#)
- P3FW_DEMO_ATQB_LEN
 - p3fw_demo.c, [121](#)
- P3FW_DEMO_RX_BUFFER
 - p3fw_demo.c, [121](#)
- P3FW_DEMO_TX_BUFFER
 - p3fw_demo.c, [121](#)
- P3FW_DEMO_UID_LEN
 - p3fw_demo.c, [121](#)
- P3FW_DIP_CFG_BAL_I2C
 - p3fw.c, [115](#)
- P3FW_DIP_CFG_BAL_SPI
 - p3fw.c, [115](#)
- P3FW_DIP_CFG_BAL_UART
 - p3fw.c, [115](#)
- P3FW_DIP_CFG_SAM_IN_X
 - p3fw.c, [115](#)
- P3FW_DIP_CFG_SAM_NO
 - p3fw.c, [115](#)
- P3FW_DIP_CFG_SAM_NON_X
 - p3fw.c, [115](#)
- P3FW_DIP_EXT_IF_ETHERNET
 - p3fw.c, [115](#)
- P3FW_DIP_EXT_IF_RS232
 - p3fw.c, [115](#)
- P3FW_DIP_EXT_IF_RS485
 - p3fw.c, [115](#)
- P3FW_DIP_EXT_IF_USB
 - p3fw.c, [116](#)
- P3FW_DIP_MODE_ACT_BOOTLOADER
 - p3fw.c, [116](#)
- P3FW_DIP_MODE_CFG_OVERWRITE
 - p3fw.c, [116](#)
- P3FW_DIP_MODE_DEMO
 - p3fw.c, [116](#)
- P3FW_DIP_MODE_PCSC
 - p3fw.c, [116](#)
- P3FW_DM_CHECK_LENGTH_EQUAL
 - Direct, [22](#)
- P3FW_DM_CHECK_LENGTH_LARGER
 - Direct, [22](#)
- P3FW_DM_CHECK_STATUS
 - Direct, [22](#)
- P3FW_DM_CID
 - Direct, [22](#)
- p3fw_dm_cid
 - Direct, [33](#)
- P3FW_DM_CID_FREE
 - Direct, [23](#)
- P3FW_DM_CID_GET_FREE
 - Direct, [23](#)
- P3FW_DM_CID_INIT
 - Direct, [23](#)

- P3FW_DM_CONTACT_CARD
Direct, [23](#)
- p3fw_dm_contact_card
Direct, [33](#)
- P3FW_DM_CONTACTCARD_ACTIVATE_-
CARD
Direct, [23](#)
- P3FW_DM_CONTACTCARD_CLOCK_-
START
Direct, [23](#)
- P3FW_DM_CONTACTCARD_CLOCK_-
STOP
Direct, [23](#)
- P3FW_DM_CONTACTCARD_COLD_RESET
Direct, [23](#)
- P3FW_DM_CONTACTCARD_DEACTIVATE_-
CARD
Direct, [24](#)
- P3FW_DM_CONTACTCARD_PPS
Direct, [24](#)
- P3FW_DM_CONTACTCARD_PRESENCE_-
CHECK
Direct, [24](#)
- P3FW_DM_CONTACTCARD_TRANSMIT_-
DATA
Direct, [24](#)
- P3FW_DM_CONTACTCARD_WARM_-
RESET
Direct, [24](#)
- p3fw_dm_get_class
Direct, [34](#)
- p3fw_dm_get_input_payload
Direct, [34](#)
- p3fw_dm_get_input_payload_length
Direct, [34](#)
- p3fw_dm_get_input_pointer_uint8_t
Direct, [34](#)
- p3fw_dm_get_input_uint16_t
Direct, [34](#)
- p3fw_dm_get_input_uint32_t
Direct, [34](#)
- p3fw_dm_get_input_uint8_t
Direct, [34](#)
- p3fw_dm_get_instruction
Direct, [35](#)
- p3fw_dm_get_output_payload
Direct, [35](#)
- p3fw_dm_get_output_pointer_uint8_t
Direct, [35](#)
- p3fw_dm_get_slot_index
Direct, [35](#)
- Direct, [35](#)
- P3FW_DM_HAL
Direct, [24](#)
- p3fw_dm_hal
Direct, [35](#)
- p3fw_dm_hal.c
aHalBuffers, [125](#)
- P3FW_DM_HAL_APP_PROT_SET
Direct, [24](#)
- P3FW_DM_HAL_EXEC_CMD
Direct, [24](#)
- P3FW_DM_HAL_GET_CFG
Direct, [25](#)
- P3FW_DM_HAL_INIT
Direct, [25](#)
- P3FW_DM_HAL_MFC_AUTH
Direct, [25](#)
- P3FW_DM_HAL_MFC_AUTH_KEY
Direct, [25](#)
- P3FW_DM_HAL_SET_CFG
Direct, [25](#)
- P3FW_DM_HAL_WAIT
Direct, [25](#)
- P3FW_DM_HAL_XCHG
Direct, [25](#)
- p3fw_dm_keystore
Direct, [35](#)
- P3FW_DM_KSTOR
Direct, [25](#)
- P3FW_DM_KSTOR_CHG_KUC
Direct, [26](#)
- P3FW_DM_KSTOR_FORMAT_KEY
Direct, [26](#)
- P3FW_DM_KSTOR_GET_CFG_STR
Direct, [26](#)
- P3FW_DM_KSTOR_GET_CONFIG
Direct, [26](#)
- P3FW_DM_KSTOR_GET_KEY
Direct, [26](#)
- P3FW_DM_KSTOR_GET_KEY_ENTRY
Direct, [26](#)
- P3FW_DM_KSTOR_GET_KUC
Direct, [26](#)
- P3FW_DM_KSTOR_INIT
Direct, [26](#)
- P3FW_DM_KSTOR_SET_CFG_STR
Direct, [27](#)
- P3FW_DM_KSTOR_SET_CONFIG
Direct, [27](#)
- P3FW_DM_KSTOR_SET_FULL_KEY

Direct, 27	Direct, 30
P3FW_DM_KSTOR_SET_KEY	p3fw_dm_l4a
Direct, 27	Direct, 36
P3FW_DM_KSTOR_SET_KEY_POS	P3FW_DM_L4A_ACT_CARD
Direct, 27	Direct, 30
P3FW_DM_KSTOR_SET_KUC	P3FW_DM_L4A_GET_PROTO_PARM
Direct, 27	Direct, 30
P3FW_DM_L3	P3FW_DM_L4A_INIT
Direct, 27	Direct, 30
p3fw_dm_l3	P3FW_DM_L4A_PPS
Direct, 35	Direct, 30
P3FW_DM_L3_ACT_CARD	P3FW_DM_L4A_RATS
Direct, 27	Direct, 30
P3FW_DM_L3_ANTICOL	P3FW_DM_RO
Direct, 28	Direct, 30
P3FW_DM_L3_GET_SER	p3fw_dm_ro
Direct, 28	Direct, 36
P3FW_DM_L3_HLTA	P3FW_DM_RO_CONF_OVER
Direct, 28	Direct, 31
P3FW_DM_L3_INIT	P3FW_DM_RO_FIELD_OFF
Direct, 28	Direct, 31
P3FW_DM_L3_REQA	P3FW_DM_RO_FIELD_ON
Direct, 28	Direct, 31
P3FW_DM_L3_SELECT	P3FW_DM_RO_FIELD_RESET
Direct, 28	Direct, 31
P3FW_DM_L3_WKUA	P3FW_DM_RO_GET_CONF
Direct, 28	Direct, 31
P3FW_DM_L3_XCHG	P3FW_DM_RO_GET_STATUS
Direct, 28	Direct, 31
P3FW_DM_L4	P3FW_DM_RO_LEDS_OFF
Direct, 29	Direct, 31
p3fw_dm_l4	P3FW_DM_RO_LEDS_ON
Direct, 36	Direct, 31
P3FW_DM_L4_DESELECT	P3FW_DM_RO_READ_REG
Direct, 29	Direct, 32
P3FW_DM_L4_GET_CFG	P3FW_DM_RO_RESET
Direct, 29	Direct, 32
P3FW_DM_L4_INIT	P3FW_DM_RO_SET_CONF
Direct, 29	Direct, 32
P3FW_DM_L4_PRES_CHECK	P3FW_DM_RO_SET_PCSC_MODE
Direct, 29	Direct, 32
P3FW_DM_L4_RESET_PROTO	P3FW_DM_RO_TEST_MODE
Direct, 29	Direct, 32
P3FW_DM_L4_SET_CFG	P3FW_DM_RO_WRITE_REG
Direct, 29	Direct, 32
P3FW_DM_L4_SET_PROTO	p3fw_dm_send_frame
Direct, 29	Direct, 36
P3FW_DM_L4_XCHG	P3FW_DM_XCHG
Direct, 30	Direct, 32
P3FW_DM_L4A	p3fw_dm_xchg

- Direct, [36](#)
- P3FW_DM_XCHG_INIT
 - Direct, [32](#)
- P3FW_DM_XCHG_L3
 - Direct, [33](#)
- P3FW_DM_XCHG_L4
 - Direct, [33](#)
- P3FW_DM_XCHG_MFC_AUTH
 - Direct, [33](#)
- P3FW_DM_XCHG_MFC_AUTH_KEY
 - Direct, [33](#)
- P3FW_DM_XCHG_PC
 - Direct, [33](#)
- P3FW_DM_XCHG_RAW
 - Direct, [33](#)
- p3fw_dm_xfer
 - Direct, [36](#)
- p3fw_dump_regs
 - p3fw.c, [116](#)
- P3FW_ERR_AND_NFO_LOOP_BFL_ERROR
 - Pegoda, [42](#)
- P3FW_ERR_AND_NFO_LOOP_CHIP - NOT_DETECTED
 - Pegoda, [42](#)
- P3FW_ERR_AND_NFO_LOOP_COM - ERROR
 - Pegoda, [42](#)
- P3FW_ERR_AND_NFO_LOOP_DONE
 - Pegoda, [42](#)
- P3FW_ERR_AND_NFO_LOOP_ERASE - FAILD
 - Pegoda, [42](#)
- P3FW_ERR_AND_NFO_LOOP_FLASH - FAILD
 - Pegoda, [42](#)
- P3FW_ERR_AND_NFO_LOOP_HAL_CAN - NOT_BE_SET
 - Pegoda, [42](#)
- P3FW_ERR_AND_NFO_LOOP_HW_ERROR
 - Pegoda, [42](#)
- P3FW_ERR_AND_NFO_LOOP_OS_ERROR
 - Pegoda, [43](#)
- P3FW_ERR_AND_NFO_LOOP_UNKNOWN - ERROR
 - Pegoda, [43](#)
- P3FW_ERR_AND_NFO_LOOP_USB_ERROR
 - Pegoda, [43](#)
- p3fw_ext_if_init_usb
 - Pegoda, [48](#)
- p3fw_ext_if_usb.c
 - Pegoda, [48](#)
- P3FW_MAX_SLOT_INDEX, [131](#)
- P3FW_NAME_USB_OFFSET, [131](#)
- P3FW_SERNUM_USB_OFFSET, [131](#)
- P3FW_USB_CONTROL_REQUEST - ABORT, [132](#)
- P3FW_USB_CONTROL_REQUEST - GET_CLOCK_FREQUENCIES, [132](#)
- P3FW_USB_CONTROL_REQUEST - GET_DATA_RATES, [132](#)
- P3FW_USB_LE_DWORD, [132](#)
- P3FW_USB_LE_WORD, [132](#)
- P3FW_USB_MAX_PACKET_SIZE, [132](#)
- p3fw_external_interface
 - Pegoda, [46](#)
- p3fw_flash.c
 - IAP, [136](#)
 - IAP_BLANK_CHECK_SECTOR, [134](#)
 - IAP_COMPARE, [134](#)
 - IAP_COPY_RAM_TO_FLASH, [134](#)
 - IAP_ERASE_SECTOR, [134](#)
 - IAP_LOCATION, [134](#)
 - IAP_PREP_SECTORS, [134](#)
 - IAP_READ_BOOT_CODE_VER, [134](#)
 - IAP_READ_DEV_SER_NUM, [134](#)
 - IAP_READ_ID, [134](#)
 - IAP_REINVOKE, [135](#)
 - IAP_SEC_29, [135](#)
 - P3FW_CFG_COUNT, [135](#)
 - P3FW_CFG_ENTRY_OFFSET, [135](#)
 - P3FW_CFG_ENTRY_SIZE, [135](#)
 - P3FW_CFG_MAX, [135](#)
 - P3FW_CFG_S1, [135](#)
 - P3FW_CFG_S2, [135](#)
 - P3FW_FLASH_BUFFER, [135](#)
 - P3FW_FLASH_BUFFER
 - p3fw_flash.c, [135](#)
 - p3fw_flash_erase_config, [48](#)
 - p3fw_flash_get_config, [48](#)
 - p3fw_flash_read_serial, [48](#)
 - p3fw_flash_set_config, [49](#)
 - p3fw_hal_buffer, [76](#)
 - rx, [76](#)
 - tx, [76](#)
 - p3fw_hw_ctrl_signal

- Hardware, [54](#)
- p3fw_hw_init
 - Hardware, [54](#)
- P3FW_HW_SIGNAL_ANTENNA
 - Hardware, [54](#)
- P3FW_HW_SIGNAL_BEEPER
 - Hardware, [54](#)
- P3FW_HW_SIGNAL_YELLOW_2
 - Hardware, [54](#)
- P3FW_HW_SIGNAL_YELLOW_3
 - Hardware, [54](#)
- P3FW_HW_SIGNAL_YELLOW_4
 - Hardware, [54](#)
- p3fw_ic
 - Pegoda, [46](#)
- p3fw_invoke_err_and_nfo_mode
 - Pegoda, [49](#)
- P3FW_KEYSTORE_NUM_KEYS
 - Pegoda, [43](#)
- P3FW_KEYSTORE_NUM_VERS
 - Pegoda, [43](#)
- P3FW_MAJOR
 - Pegoda, [43](#)
- P3FW_MAX_ATQB_LEN
 - Pegoda, [43](#)
- P3FW_MAX_ATS_LEN
 - Pegoda, [43](#)
- P3FW_MAX_CCID_BUFFER_LEN
 - Pegoda, [44](#)
- P3FW_MAX_CONTACT_SLOTS
 - Pegoda, [44](#)
- P3FW_MAX_CONTACTLESS_SLOTS
 - Pegoda, [44](#)
- P3FW_MAX_HAL_RX_BUFFER
 - Pegoda, [44](#)
- P3FW_MAX_HAL_TX_BUFFER
 - Pegoda, [44](#)
- P3FW_MAX_KEY_LEN
 - Pegoda, [44](#)
- P3FW_MAX_SLOT_INDEX
 - p3fw_ext_if_usb.c, [131](#)
- P3FW_MAX_UID_A_LEN
 - Pegoda, [44](#)
- P3FW_MAX_UID_B_LEN
 - Pegoda, [44](#)
- P3FW_MINOR
 - Pegoda, [45](#)
- p3fw_mode
 - Pegoda, [46](#)
- P3FW_NAME_USB_OFFSET
 - p3fw_ext_if_usb.c, [131](#)
- P3FW_PCSC_FSDI
 - Pegoda, [45](#)
- p3fw_pcsc_mode
 - Pegoda, [47](#)
- P3FW_PCSC_PROTO_PARAMS_LEN_ - T0
 - Pegoda, [45](#)
- P3FW_PCSC_PROTO_PARAMS_LEN_ - T1
 - Pegoda, [45](#)
- p3fw_pcsc_protocol
 - Pegoda, [47](#)
- p3fw_pcsc_send_apdu
 - PCSCs, [55](#)
- p3fw_pcsc_std_ext
 - PCSCs, [56](#)
- p3fw_pcsc_std_ext_do_auth
 - PCSCs, [56](#)
- p3fw_pcsc_std_ext_prepare_l3_card
 - PCSCs, [56](#)
- p3fw_sam
 - Pegoda, [47](#)
- p3fw_samt1.c
 - BITBAND_PERI, [140](#)
 - BITBAND_PERI_BASE, [140](#)
 - BITBAND_PERI_REF, [140](#)
 - EINT3_IRQHandler, [143](#)
 - P2_FW_SAM_CLOCK_LEN_NS, [140](#)
 - P2_FW_SAM_CONVERSION_DIRECT, [140](#)
 - P2_FW_SAM_CONVERSION_INVERSE, [141](#)
 - P2_FW_SAM_CONVERSION_UNKNOWN, [141](#)
 - P2_FW_SAM_DEBUG, [141](#)
 - P2_FW_SAM_DEFAULT_BWI, [141](#)
 - P2_FW_SAM_MAX_ATR_SIZE, [141](#)
 - P2_FW_SAM_MAX_REC_DATALEN, [141](#)
 - P2_FW_SAM_MODE_IDLE, [141](#)
 - P2_FW_SAM_MODE_POWER_OFF, [141](#)
 - P2_FW_SAM_MODE_RECEIVE, [141](#)
 - P2_FW_SAM_MODE_RESET, [141](#)
 - P2_FW_SAM_MODE_SEND, [141](#)
 - P2_FW_SAM_MODE_WAIT_START_ - BIT, [142](#)
 - P2_FW_SAM_PWM_CH0_MATCH_ - VALUE, [142](#)

- P2_FW_SAM_PWM_CH1_MATCH_VALUE, 142
- P2_FW_SAM_T1_CLOCK_CONFIG, 142
- p2_fw_sam_t1_deinit, 143
- p2_fw_sam_t1_get_atr, 143
- p2_fw_sam_t1_get_cont_tim, 143
- p2_fw_sam_t1_get_timing_mode, 143
- p2_fw_sam_t1_init, 143
- p2_fw_sam_t1_is_busy, 143
- p2_fw_sam_t1_is_power_off, 143
- p2_fw_sam_t1_is_sam_inserted, 143
- p2_fw_sam_t1_pps, 144
- p2_fw_sam_t1_prepare_pps, 144
- p2_fw_sam_t1_receive, 144
- p2_fw_sam_t1_send, 144
- p2_fw_sam_t1_set_bwi_cwi, 144
- p2_fw_sam_t1_set_etu, 144
- p2_fw_sam_t1_set_my_debug, 144
- p2_fw_sam_t1_set_rec_extraGuardTime, 144
- p2_fw_sam_t1_set_timing_mode, 144
- p2_fw_sam_t1_start, 144
- p2_fw_sam_t1_warm_reset, 144
- P2_FW_SAM_TIMER_PRESCALE_VALUE, 142
- T1_BYTE_WAIT_TIME, 142
- TIMER0_IR, 142
- TIMER0_IR_MR0, 142
- TIMER0_IRQHandler, 145
- TIMER0_TCR, 142
- TIMER0_TCR_ENABLE, 142
- TIMER0_TCR_RESET, 143
- TIMER2_IRQHandler, 145
- p3fw_samt1.h
 - EINT3_IRQHandler, 104
 - P2_FW_SAM_FRAME_APDU, 102
 - P2_FW_SAM_FRAME_T1, 102
 - P2_FW_SAM_MODE_PPS, 102
 - P2_FW_SAM_MODE_X, 102
 - p2_fw_sam_t1_deinit, 104
 - p2_fw_sam_t1_get_atr, 104
 - p2_fw_sam_t1_get_cont_tim, 104
 - p2_fw_sam_t1_get_timing_mode, 104
 - p2_fw_sam_t1_init, 104
 - p2_fw_sam_t1_is_busy, 105
 - p2_fw_sam_t1_is_power_off, 105
 - p2_fw_sam_t1_is_sam_inserted, 105
 - p2_fw_sam_t1_pps, 105
 - p2_fw_sam_t1_prepare_pps, 105
 - p2_fw_sam_t1_receive, 105
 - p2_fw_sam_t1_send, 105
 - p2_fw_sam_t1_set_bwi_cwi, 105
 - p2_fw_sam_t1_set_etu, 105
 - p2_fw_sam_t1_set_my_debug, 105
 - p2_fw_sam_t1_set_rec_extraGuardTime, 105
 - p2_fw_sam_t1_set_timing_mode, 106
 - p2_fw_sam_t1_start, 106
 - p2_fw_sam_t1_warm_reset, 106
 - P2_FW_TIMING_MODE_COM, 103
 - P2_FW_TIMING_MODE_FDT, 103
 - P2_FW_TIMING_MODE_NONE, 103
 - p3fw_samt1_deinit, 103
 - p3fw_samt1_exchange, 106
 - p3fw_samt1_get_cont_tim, 103
 - p3fw_samt1_get_timing_mode, 103
 - p3fw_samt1_getATR, 103
 - p3fw_samt1_init, 106
 - p3fw_samt1_is_busy, 103
 - p3fw_samt1_is_sam_inserted, 103
 - p3fw_samt1_power_on, 106
 - p3fw_samt1_receive, 103
 - p3fw_samt1_send, 103
 - p3fw_samt1_set_etu, 104
 - p3fw_samt1_set_param, 106
 - p3fw_samt1_set_timing_mode, 104
 - p3fw_samt1_start, 104
 - p3fw_samt1_warm_reset, 104
 - TIMER0_IRQHandler, 106
 - TIMER2_IRQHandler, 106
- p3fw_samt1_deinit
- p3fw_samt1.h, 103
- p3fw_samt1_exchange
- p3fw_samt1.h, 106
- p3fw_samt1_wrap.c, 146
- p3fw_samt1_get_cont_tim
- p3fw_samt1.h, 103
- p3fw_samt1_get_timing_mode
- p3fw_samt1.h, 103
- p3fw_samt1_getATR
- p3fw_samt1.h, 103
- p3fw_samt1_init
- p3fw_samt1.h, 106
- p3fw_samt1_wrap.c, 146
- p3fw_samt1_is_busy
- p3fw_samt1.h, 103
- p3fw_samt1_is_sam_inserted
- p3fw_samt1.h, 103
- p3fw_samt1_power_on

- p3fw_samt1.h, 106
 - p3fw_samt1_wrap.c, 146
- p3fw_samt1_receive
 - p3fw_samt1.h, 103
- p3fw_samt1_send
 - p3fw_samt1.h, 103
- p3fw_samt1_set_etu
 - p3fw_samt1.h, 104
- p3fw_samt1_set_param
 - p3fw_samt1.h, 106
 - p3fw_samt1_wrap.c, 146
- p3fw_samt1_set_timing_mode
 - p3fw_samt1.h, 104
- p3fw_samt1_start
 - p3fw_samt1.h, 104
- p3fw_samt1_warm_reset
 - p3fw_samt1.h, 104
- p3fw_samt1_wrap.c
 - p3fw_samt1_exchange, 146
 - p3fw_samt1_init, 146
 - p3fw_samt1_power_on, 146
 - p3fw_samt1_set_param, 146
- P3FW_SERNUM_USB_OFFSET
 - p3fw_ext_if_usb.c, 131
- p3fw_slot_add_l3a_card
 - Pegoda, 50
- p3fw_slot_add_l3b_card
 - Pegoda, 50
- p3fw_slot_add_l4_card
 - Pegoda, 50
- p3fw_slot_get_atr
 - Pegoda, 50
- p3fw_slot_remove_cl_card
 - Pegoda, 51
- p3fw_slot_reset_all_slots
 - Pegoda, 51
- p3fw_slot_types
 - Pegoda, 47
- p3fw_slots.c
 - aHalBuffers, 147
 - P3FW_TEMP_BUFF_LEN, 147
- p3fw_slots_init
 - Pegoda, 51
- p3fw_task_ccid_execute
 - Pegoda, 51
- p3fw_task_demo_mode
 - Pegoda, 52
- p3fw_task_poll_and_activate
 - Pegoda, 52
- P3FW_TASK_START_ERROR
 - p3fw.c, 116
- P3FW_TASK_START_EXECUTE
 - p3fw.c, 116
- P3FW_TASK_START_POLL
 - p3fw.c, 116
- P3FW_TEMP_BUFF_LEN
 - p3fw_slots.c, 147
- p3fw_timing_init
 - Pegoda, 52
- p3fw_timing_start
 - Pegoda, 52
- p3fw_timing_stop
 - Pegoda, 53
- P3FW_USB_BULK_IN_EP
 - Pegoda, 45
- P3FW_USB_BULK_OUT_EP
 - Pegoda, 45
- P3FW_USB_CONTROL_REQUEST_ABORT
 - p3fw_ext_if_usb.c, 132
- P3FW_USB_CONTROL_REQUEST_GET_-
CLOCK_FREQUENCIES
 - p3fw_ext_if_usb.c, 132
- P3FW_USB_CONTROL_REQUEST_GET_-
DATA_RATES
 - p3fw_ext_if_usb.c, 132
- P3FW_USB_INT_IN_EP
 - Pegoda, 45
- P3FW_USB_LE_DWORD
 - p3fw_ext_if_usb.c, 132
- P3FW_USB_LE_WORD
 - p3fw_ext_if_usb.c, 132
- P3FW_USB_MAX_PACKET_SIZE
 - p3fw_ext_if_usb.c, 132
- parityCount
 - p2_fw_SAM_ctrl_, 66
- pBal
 - p3fw_data, 74
- pcsc
 - p3fw_data, 74
- PCSCs, 55
 - p3fw_pcsc_send_apdu, 55
 - p3fw_pcsc_std_ext, 56
 - p3fw_pcsc_std_ext_do_auth, 56
 - p3fw_pcsc_std_ext_prepare_l3_card, 56
- Pegoda
 - p3fw, 53
 - P3FW_BAL_I2C, 46
 - P3FW_BAL_SPI, 46
 - P3FW_BAL_UART, 46

- P3FW_EXT_IF_ETHERNET, 46
- P3FW_EXT_IF_RS232, 46
- P3FW_EXT_IF_RS485, 46
- P3FW_EXT_IF_USB, 46
- P3FW_IC_RC523, 46
- P3FW_IC_RC663, 46
- P3FW_IC_SAM, 46
- P3FW_MODE_ACTIVATE_BOOTLOADER, 47
- P3FW_MODE_CFG_OVERWRITE, 47
- P3FW_MODE_DEMO, 46
- P3FW_MODE_PCSC, 46
- P3FW_PCSC_MODE_DIRECT, 47
- P3FW_PCSC_MODE_NORMAL, 47
- P3FW_PCSC_PROTOCOL_RAW, 47
- P3FW_PCSC_PROTOCOL_T0, 47
- P3FW_PCSC_PROTOCOL_T1, 47
- P3FW_PCSC_PROTOCOL_UNKNOWN, 47
- P3FW_SAM_IN_X, 47
- P3FW_SAM_NO, 47
- P3FW_SAM_NON_X, 47
- P3FW_SLOT_EMPTY, 47
- P3FW_SLOT_ISO1444L3A_CARD, 48
- P3FW_SLOT_ISO1444L3B_CARD, 48
- P3FW_SLOT_ISO1444L4_CARD, 48
- P3FW_SLOT_SAM, 47
- p3fw_bal_configuration, 46
- P3FW_BUILD, 40
- p3fw_cc_slot, 45
- P3FW_CFG_BOOTLOADER_ACTIVE, 40
- P3FW_CFG_BOOTLOADER_VERSION, 40
- P3FW_CFG_CONT_TIMING, 41
- P3FW_CFG_GET_CONT_TIMING, 41
- P3FW_CFG_MAX_SLOTS, 41
- P3FW_CFG_SET_DIP_SWITCHES, 41
- P3FW_CHECK_STATUS, 41
- p3fw_cl_slot, 45
- P3FW_CONTACT_SLOTS_MASK, 41
- P3FW_ERR_AND_NFO_LOOP_BFL_-
ERROR, 42
- P3FW_ERR_AND_NFO_LOOP_CHIP_-
NOT_DETECTED, 42
- P3FW_ERR_AND_NFO_LOOP_COM_-
ERROR, 42
- P3FW_ERR_AND_NFO_LOOP_DONE, 42
- P3FW_ERR_AND_NFO_LOOP_ERASE_-
FAILED, 42
- P3FW_ERR_AND_NFO_LOOP_FLASH_-
FAILED, 42
- P3FW_ERR_AND_NFO_LOOP_HAL_-
CAN_NOT_BE_SET, 42
- P3FW_ERR_AND_NFO_LOOP_HW_-
ERROR, 42
- P3FW_ERR_AND_NFO_LOOP_OS_-
ERROR, 43
- P3FW_ERR_AND_NFO_LOOP_UNKNOWN_-
ERROR, 43
- P3FW_ERR_AND_NFO_LOOP_USB_-
ERROR, 43
- p3fw_ext_if_init_usb, 48
- p3fw_external_interface, 46
- p3fw_flash_erase_config, 48
- p3fw_flash_get_config, 48
- p3fw_flash_read_serial, 48
- p3fw_flash_set_config, 49
- p3fw_ic, 46
- p3fw_invoke_err_and_nfo_mode, 49
- P3FW_KEYSTORE_NUM_KEYS, 43
- P3FW_KEYSTORE_NUM_VERS, 43
- P3FW_MAJOR, 43
- P3FW_MAX_ATQB_LEN, 43
- P3FW_MAX_ATS_LEN, 43
- P3FW_MAX_CCID_BUFFER_LEN, 44
- P3FW_MAX_CONTACT_SLOTS, 44
- P3FW_MAX_CONTACTLESS_SLOTS, 44
- P3FW_MAX_HAL_RX_BUFFER, 44
- P3FW_MAX_HAL_TX_BUFFER, 44
- P3FW_MAX_KEY_LEN, 44
- P3FW_MAX_UID_A_LEN, 44
- P3FW_MAX_UID_B_LEN, 44
- P3FW_MINOR, 45
- p3fw_mode, 46
- P3FW_PCSC_FSDI, 45
- p3fw_pcsc_mode, 47
- P3FW_PCSC_PROTO_PARAMS_LEN_-
T0, 45

- P3FW_PCSC_PROTO_PARAMS_LEN- p2_fw_SAM_ctrl_, 66
- T1, 45
- p3fw_pcsc_protocol, 47
- p3fw_sam, 47
- p3fw_slot_add_l3a_card, 50
- p3fw_slot_add_l3b_card, 50
- p3fw_slot_add_l4_card, 50
- p3fw_slot_get_atr, 50
- p3fw_slot_remove_cl_card, 51
- p3fw_slot_reset_all_slots, 51
- p3fw_slot_types, 47
- p3fw_slots_init, 51
- p3fw_task_ccid_execute, 51
- p3fw_task_demo_mode, 52
- p3fw_task_poll_and_activate, 52
- p3fw_timing_init, 52
- p3fw_timing_start, 52
- p3fw_timing_stop, 53
- P3FW_USB_BULK_IN_EP, 45
- P3FW_USB_BULK_OUT_EP, 45
- P3FW_USB_INT_IN_EP, 45
- pHal
 - _p3fw_cl_slot, 61
- recExtraGuardTime
 - p2_fw_SAM_ctrl_, 66
- rx
 - p3fw_hal_buffer, 76
- sam_t1_param, 76
 - BWL_CWI, 77
 - ClockStop, 77
 - FI_DI, 77
 - GuardTime, 77
 - IFSC, 77
- sBall2c
 - p3fw_data, 74
- sBalSerial
 - p3fw_data, 74
- sBalSpi
 - p3fw_data, 74
- sCidManager
 - p3fw_data, 74
- send
 - p3fw_data, 74
- send_data
 - p2_fw_SAM_ctrl_, 66
- sendLen
 - p2_fw_SAM_ctrl_, 66
- sendSeqData
 - p2_fw_SAM_ctrl_, 66
 - SessionATR
 - p2_fw_SAM_ctrl_, 66
 - SessionATR_Size
 - p2_fw_SAM_ctrl_, 66
 - sISO14443L3a
 - _p3fw_cl_slot, 61
 - sISO14443L3b
 - _p3fw_cl_slot, 62
 - sISO14443L4
 - _p3fw_cl_slot, 62
 - sISO14443L4a
 - _p3fw_cl_slot, 62
 - sKeyEntry
 - p3fw_data, 75
 - sKeyKUC
 - p3fw_data, 75
 - sKeyStore
 - p3fw_data, 75
 - sKeyVersion
 - p3fw_data, 75
 - slots
 - p3fw_data, 75
 - sMfc
 - _p3fw_cl_slot, 62
 - sMful
 - _p3fw_cl_slot, 62
 - sMifare
 - _p3fw_cl_slot, 62
 - sMultexBal
 - p3fw_data, 75
 - sQueueCCIDExec
 - p3fw_data, 75
 - src/debug_frmwrk.c, 107
 - src/p3fw.c, 114
 - src/p3fw_bsp.c, 117
 - src/p3fw_ccid.c, 117
 - src/p3fw_ccid_bh.c, 119
 - src/p3fw_ccid_th.c, 120
 - src/p3fw_demo.c, 120
 - src/p3fw_dm.c, 122
 - src/p3fw_dm_cc.c, 123
 - src/p3fw_dm_cid.c, 124
 - src/p3fw_dm_hal.c, 124
 - src/p3fw_dm_keystore.c, 125
 - src/p3fw_dm_l3.c, 126
 - src/p3fw_dm_l4.c, 127
 - src/p3fw_dm_l4a.c, 127
 - src/p3fw_dm_ro.c, 128
 - src/p3fw_dm_xchg.c, 129

- src/p3fw_error_mode.c, 130
- src/p3fw_ext_if_usb.c, 130
- src/p3fw_flash.c, 132
- src/p3fw_pcsc.c, 136
- src/p3fw_pcsc_mem_cards.c, 137
- src/p3fw_poll.c, 137
- src/p3fw_samt1.c, 138
- src/p3fw_samt1_wrap.c, 145
- src/p3fw_slots.c, 146
- src/p3fw_timing.c, 148
- state
 - cpot_atr_frame, 63
- t0
 - p2_fw_SAM_ctrl_, 66
- T1_BYTE_WAIT_TIME
 - p3fw_samt1.c, 142
- TAi
 - cpot_atr_frame, 63
- TBi
 - cpot_atr_frame, 63
- TCi
 - cpot_atr_frame, 63
- TCK
 - cpot_atr_frame, 63
- TDi
 - cpot_atr_frame, 63
- TIMER0_IR
 - p3fw_samt1.c, 142
- TIMER0_IR_MR0
 - p3fw_samt1.c, 142
- TIMER0_IRQHandler
 - p3fw_samt1.c, 145
 - p3fw_samt1.h, 106
- TIMER0_TCR
 - p3fw_samt1.c, 142
- TIMER0_TCR_ENABLE
 - p3fw_samt1.c, 142
- TIMER0_TCR_RESET
 - p3fw_samt1.c, 143
- TIMER2_IRQHandler
 - p3fw_samt1.c, 145
 - p3fw_samt1.h, 106
- timingMode
 - p2_fw_SAM_ctrl_, 66
- tmpByteWait_time
 - p2_fw_SAM_ctrl_, 66
- tx
 - p3fw_hal_buffer, 76
- UARTGetChar
 - debug_frmwrk.c, 109
 - debug_frmwrk.h, 83
- UARTPutChar
 - debug_frmwrk.c, 110
 - debug_frmwrk.h, 84
- UARTPutDec
 - debug_frmwrk.c, 110
 - debug_frmwrk.h, 84
- UARTPutDec16
 - debug_frmwrk.c, 110
 - debug_frmwrk.h, 84
- UARTPutDec32
 - debug_frmwrk.c, 110
 - debug_frmwrk.h, 85
- UARTPutHex
 - debug_frmwrk.c, 111
 - debug_frmwrk.h, 85
- UARTPutHex16
 - debug_frmwrk.c, 111
 - debug_frmwrk.h, 85
- UARTPutHex32
 - debug_frmwrk.c, 111
 - debug_frmwrk.h, 86
- UARTPutHexMulti
 - debug_frmwrk.c, 112
 - debug_frmwrk.h, 86
- UARTPuts
 - debug_frmwrk.c, 112
 - debug_frmwrk.h, 86
- UARTPuts_
 - debug_frmwrk.c, 112
 - debug_frmwrk.h, 86