

Locking

This file explains the locking and exclusion scheme used in the PCCARD and PCMCIA subsystems.

A) Overview, Locking Hierarchy:

pcmcia_socket_list_rwlock

- protects only the list of sockets
- skt_mutex
 - serializes card insert / ejection
- ops_mutex
 - serializes socket operation

B) Exclusion

The following functions and callbacks to struct pcmcia_socket must be called with "skt_mutex" held:

```
socket_detect_change()
send_event()
socket_reset()
socket_shutdown()
socket_setup()
socket_remove()
socket_insert()
socket_early_resume()
socket_late_resume()
socket_resume()
socket_suspend()

struct pcmcia_callback *callback
```

The following functions and callbacks to struct pcmcia_socket must be called with "ops_mutex" held:

```
socket_reset()
socket_setup()

struct pccard_operations      *ops
struct pccard_resource_ops    *resource_ops;
```

Note that send_event() and struct pcmcia_callback *callback must not be called with "ops_mutex" held.

C) Protection

1. Global Data:

struct list_head pcmcia_socket_list;
protected by pcmcia_socket_list_rwlock;

2. Per-Socket Data:

The resource_ops and their data are protected by ops_mutex.

The "main" struct pcmcia_socket is protected as follows (read-only fields or single-use fields not mentioned):

- by pcmcia_socket_list_rwlock

```
struct list_head    socket_list;
```
- by thread_lock:

```
unsigned int        thread_events;
```
- by skt_mutex:

```
u_int               suspended_state;
void                (*tune_bridge);
struct pcmcia_callback *callback;
int                 resume_status;
```
- by ops_mutex:

```
socket_state_t      socket;
```

u_int	state;
u_short	lock_count;
pccard_mem_map	cis_mem;
void __iomem	*cis_virt;
struct { }	irq;
io_window_t	io[];
pccard_mem_map	win[];
struct list_head	cis_cache;
size_t	fake_cis_len;
u8	*fake_cis;
u_int	irq_mask;
void	(*zoom_video);
int	(*power_hook);
u8	resource...;
struct list_head	devices_list;
u8	device_count;
struct	pcmcia_state;

3. Per PCMCIA-device Data:

The "main" struct `pcmcia_device` is protected as follows (read-only fields or single-use fields not mentioned):

- by `pcmcia_socket->ops_mutex`:

struct list_head	socket_device_list;
struct config_t	*function_config;
u16	_irq:1;
u16	_io:1;
u16	_win:4;
u16	_locked:1;
u16	allow_func_id_match:1;
u16	suspended:1;
u16	_removed:1;

- by the PCMCIA driver:

io_req_t	io;
irq_req_t	irq;
config_req_t	conf;
window_handle_t	win;