

Experimental PREEMPT_RT Linux Usage in Railway Interlocking System

Andrey Fedotov
NICEVT, Moscow

June, 26th, 2014 - RTS-LIKE – Pisa

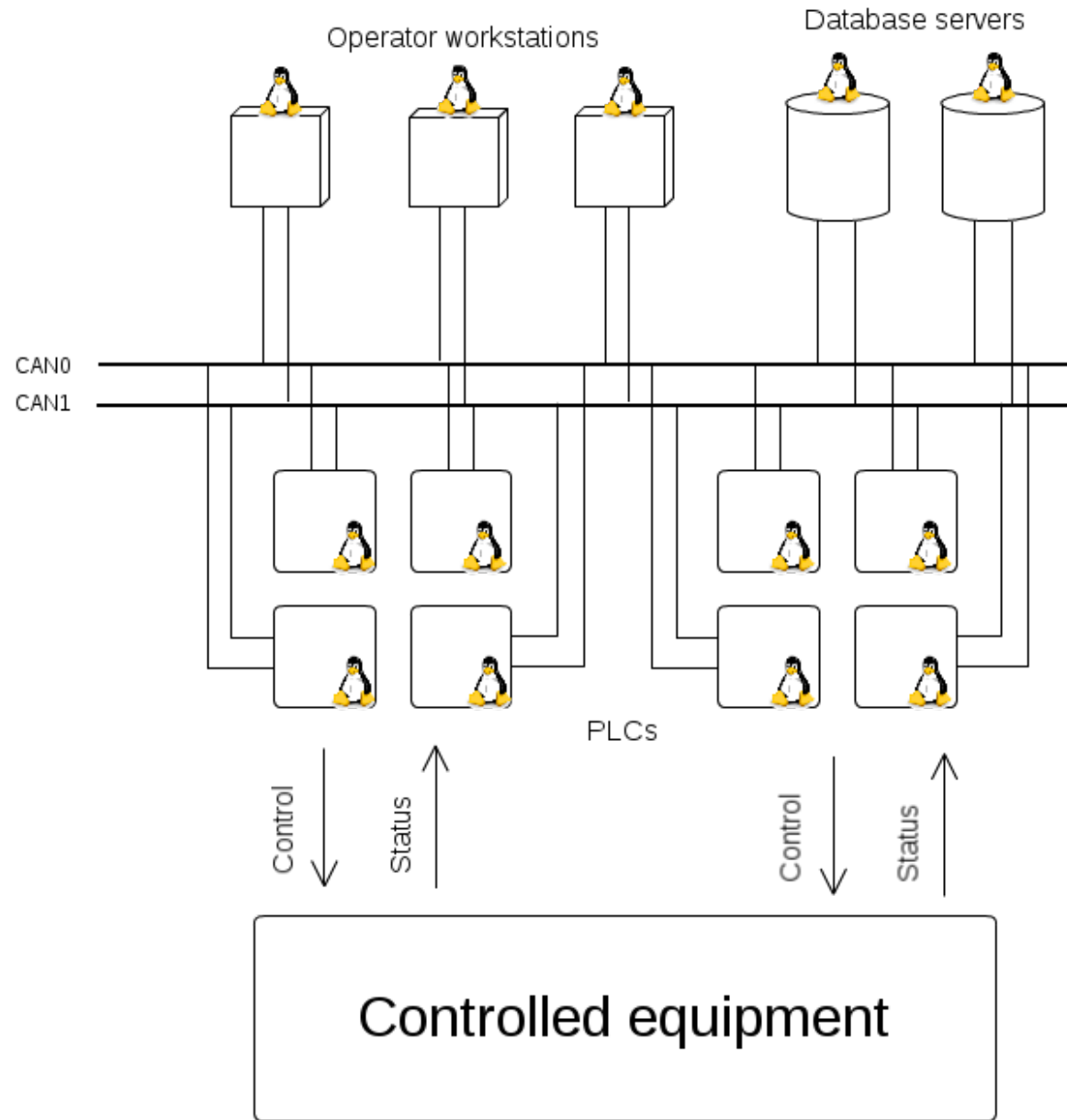
Station Scheme



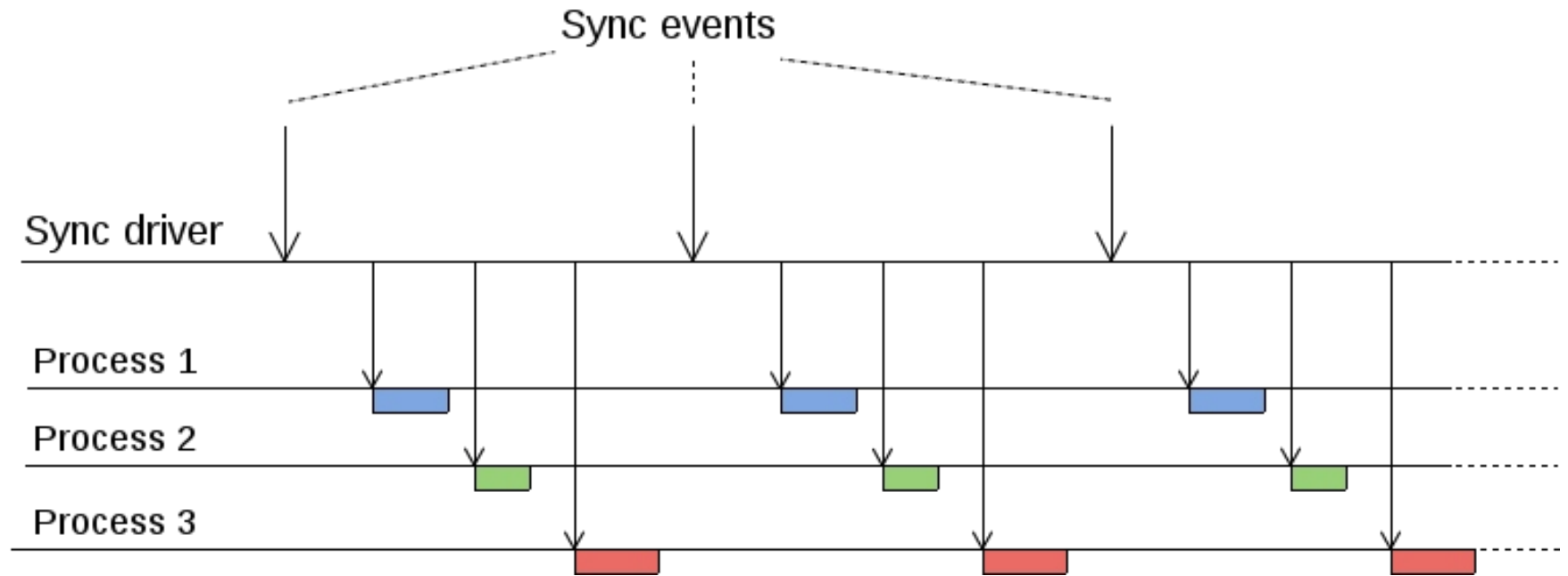
Interlocking System

- Obtains trains positions
- Controls switches
- Controls signals
- Generates control command to trains
- Performs service tasks

System Structure



Processes Synchronization



Main Cycle Of Sync Driver

```
static int main_cycle(void* data)
{
    s32 dev;

    DECLARE_WAITQUEUE(wait, current);
    add_wait_queue(&myevent_waitqueue, &wait);

    while( !kthread_should_stop() ) {
        set_current_state(TASK_INTERRUPTIBLE);
        schedule();

        // ...

        // Unblock chardevs read()
        for ( dev = 0; dev < MAX_CHARDEVS; dev++ ) {

            usleep_range(timeout_us_sync_read[dev],
                        timeout_us_sync_read[dev] + TIMEOUT_US_LIMIT);

            device_descriptor[dev]->data_to_userspace_available = 1;
            wake_up_interruptible(&device_descriptor[dev]->inq);
        }
    }

    // ...

    return 0;
}
```

Underlying System

- Requirements
 - Precision of unblocking moments
 - Kernel-space and userspace threads priorities
 - Deterministic execution time for processes under given priorities
- Existing Problems
 - Network packets processing determinism
 - Safe timing isolation of applications in mixed criticality systems