Lab 2: Lab session today
... and ..
how not to implement shared memory IPC

Lecture 5, Part X: Lab 2 drop-in session and bug
Prof. Robert N. M. Watson
2023-2024
Additional lab time this afternoon

• Lecture 5 will take a bit over an hour (but hopefully not too much more)
• We have the Intel Lab booked for this afternoon in case you have questions about Lab 2
• Some number of us will head up to the lab when this lecture wraps up – feel free to drop in and chat

• Also FYI
  • Aiming to get Lab 1 feedback to ACS/Part III students today (tonight?), and Part II students tomorrow (night?)
  • It won’t be earth shattering, but hopefully will assist in honing your presentation and discussion of results
sender_shm()

while (write_sofar < totalsize) {
    while (shmem_metadata_ptr->sm_datapresent == 1) {
        if (pthread_cond_wait(
            &shmem_metadata_ptr->sm_cond_empty,
            &shmem_metadata_ptr->sm_mutex) < 0)
            xo_err(EX_OSERR, "pthread_cond_wait");
    }
    const size_t bytes_to_write = min(buffersize, totalsize - write_sofar);
    memcpy(sap->sa_buffer, shmem_buffer_ptr, bytes_to_write);
    write_sofar += bytes_to_write;
    shmem_metadata_ptr->sm_datapresent = 1;
    if (pthread_cond_signal(&shmem_metadata_ptr->sm_cond_full) < 0)
        xo_err(EX_OSERR, "pthread_cond_signal");
}
while (read_sofar < totalsize) {
    while (shmem_metadata_ptr->sm_datapresent == 0) {
        if (pthread_cond_wait(
            &shmem_metadata_ptr->sm_cond_full,
            &shmem_metadata_ptr->sm_mutex) < 0)
            xo_err(EX_OSERR, "pthread_cond_wait");
    }
    const size_t bytes_to_read = min(buffersize, totalsize - read_sofar);
    memcpy(shmem_buffer_ptr, buf, bytes_to_read);
    read_sofar += bytes_to_read;
    shmem_metadata_ptr->sm_datapresent = 0;
    if (pthread_cond_signal(&shmem_metadata_ptr->sm_cond_empty) < 0)
        xo_err(EX_OSERR, "pthread_cond_signal");
}
Upshot

• The IPC implementation we’ve provided may not function quite as desired …
• … but it does [both conceptually and empirically] behave the same way with respect to architecture and microarchitecture …
• … at least when you use the tools we’ve provided, and framed in the way that we have.
• So do continue your labs despite this mixed news [for us much more than for you].
• Thanks to Henry Batchelor for spotting this!