组会报告

徐益

2018 年 5 月 23 日

1 本周工作内容

1. 尝试扩大 DPDK 的中 mbuf
2. 实现分块传输和流量控制问题
3. 处理 makefile 相关问题

2 尝试扩大 DPDK 的中 mbuf

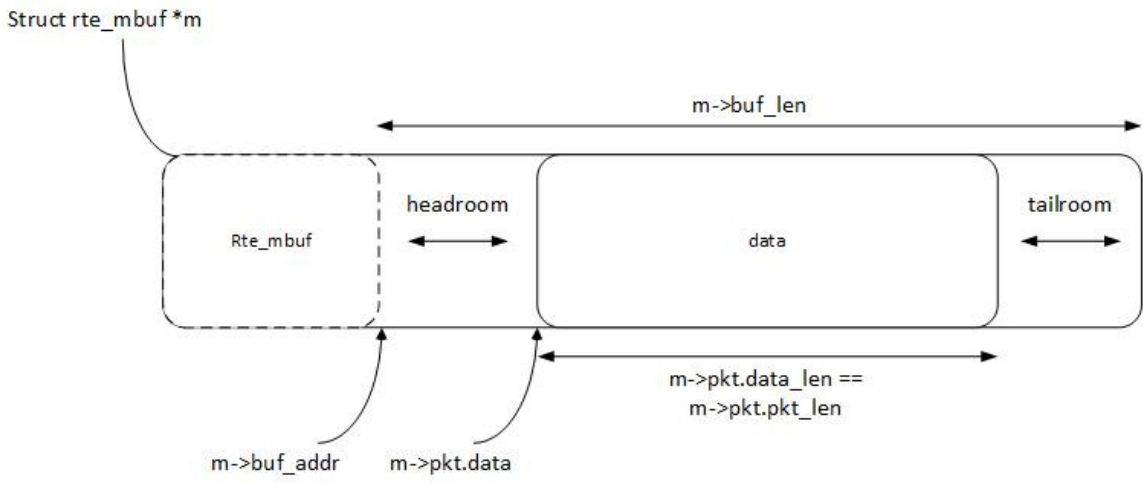

图 1: rtl_mbuf 结构

```
/**
* @param name
*   The name of the mbuf pool.
* @param n
*   The number of elements in the mbuf pool. The optimum size (in terms
*   of memory usage) for a mempool is when n is a power of two minus one:
*   n = (2^q - 1).
* @param cache_size
*   Size of the per-core object cache. See rte_mempool_create() for
*   details.
* @param priv_size
*   Size of application private are between the rte_mbuf structure
*   and the data buffer. This value must be aligned to RTE_MBUF_PRIV_ALIGN.
* @param data_room_size
*   Size of data buffer in each mbuf, including RTE_PKTMBUF_HEADROOM.
```

```
16 * @param socket_id
17 *   The socket identifier where the memory should be allocated. The
18 *   value can be *SOCKET_ID_ANY* if there is no NUMA constraint for the
19 *   reserved zone.
20 */
21 struct rte_mempool *
22 rte_pktmbuf_pool_create(const char *name, unsigned n,
23         unsigned cache_size, uint16_t priv_size, uint16_t data_room_size,
24         int socket_id);
```

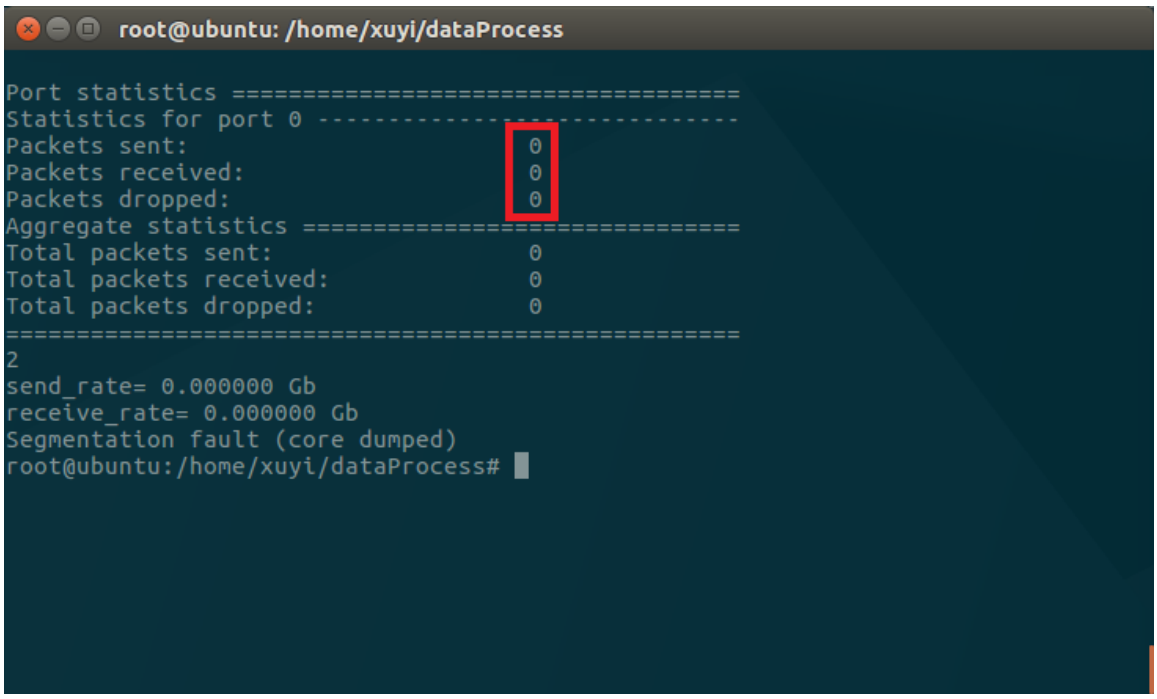

图 2: 仅扩大 MBUF_SIZE

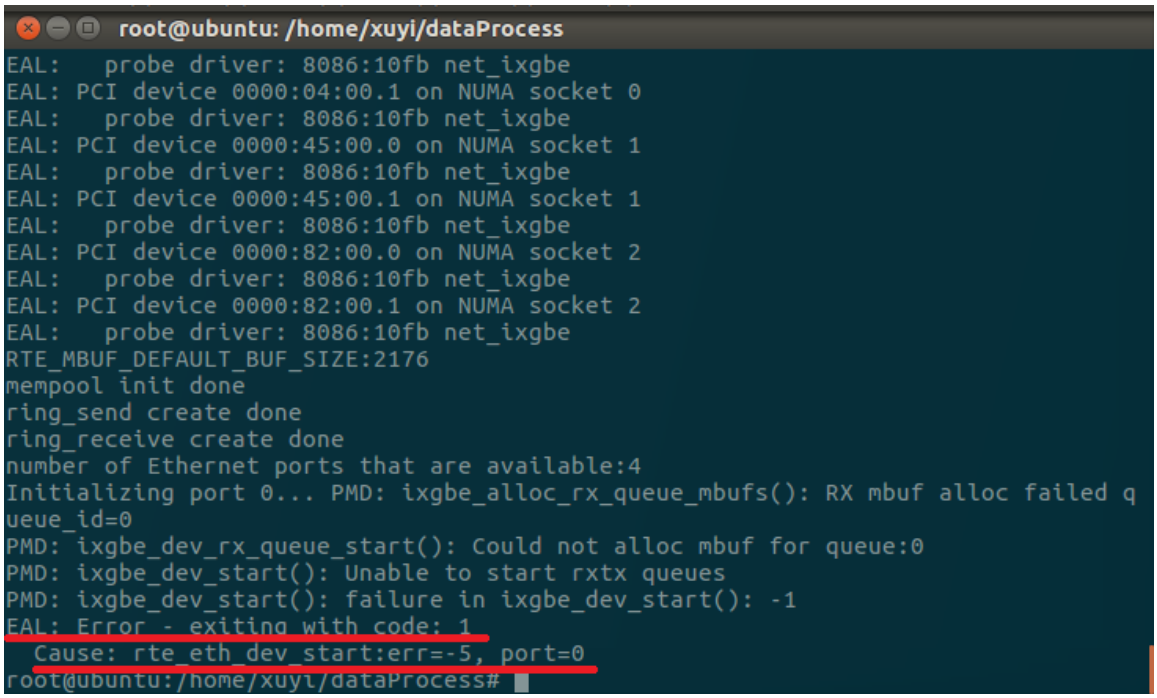

图 3: 扩大 MBUF_SIZE 的同时缩小 NB_MBUF

3 分段传输方案

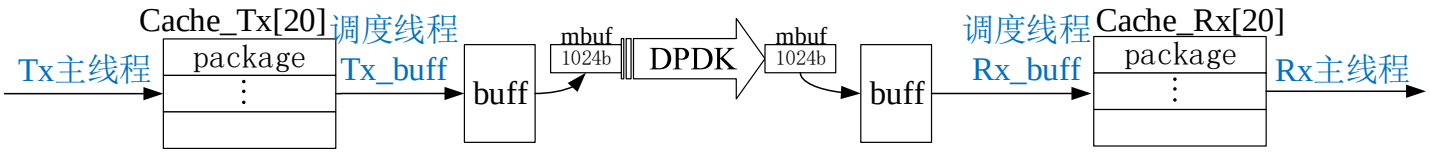

图 4: 分段传输方案系统结构

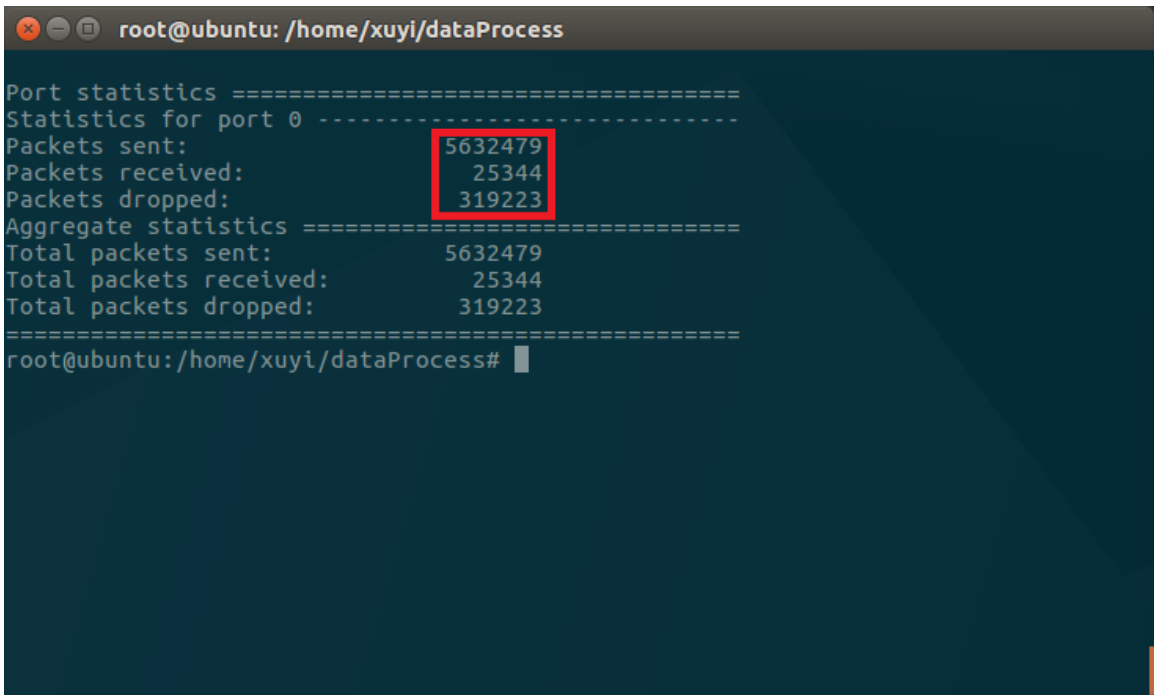

图 5: 未进行流量控制

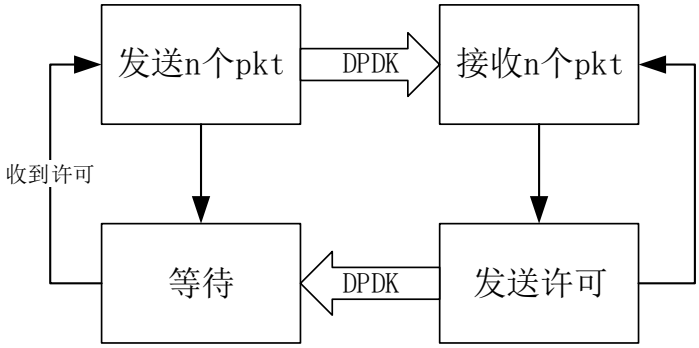

图 6: 流量控制方案

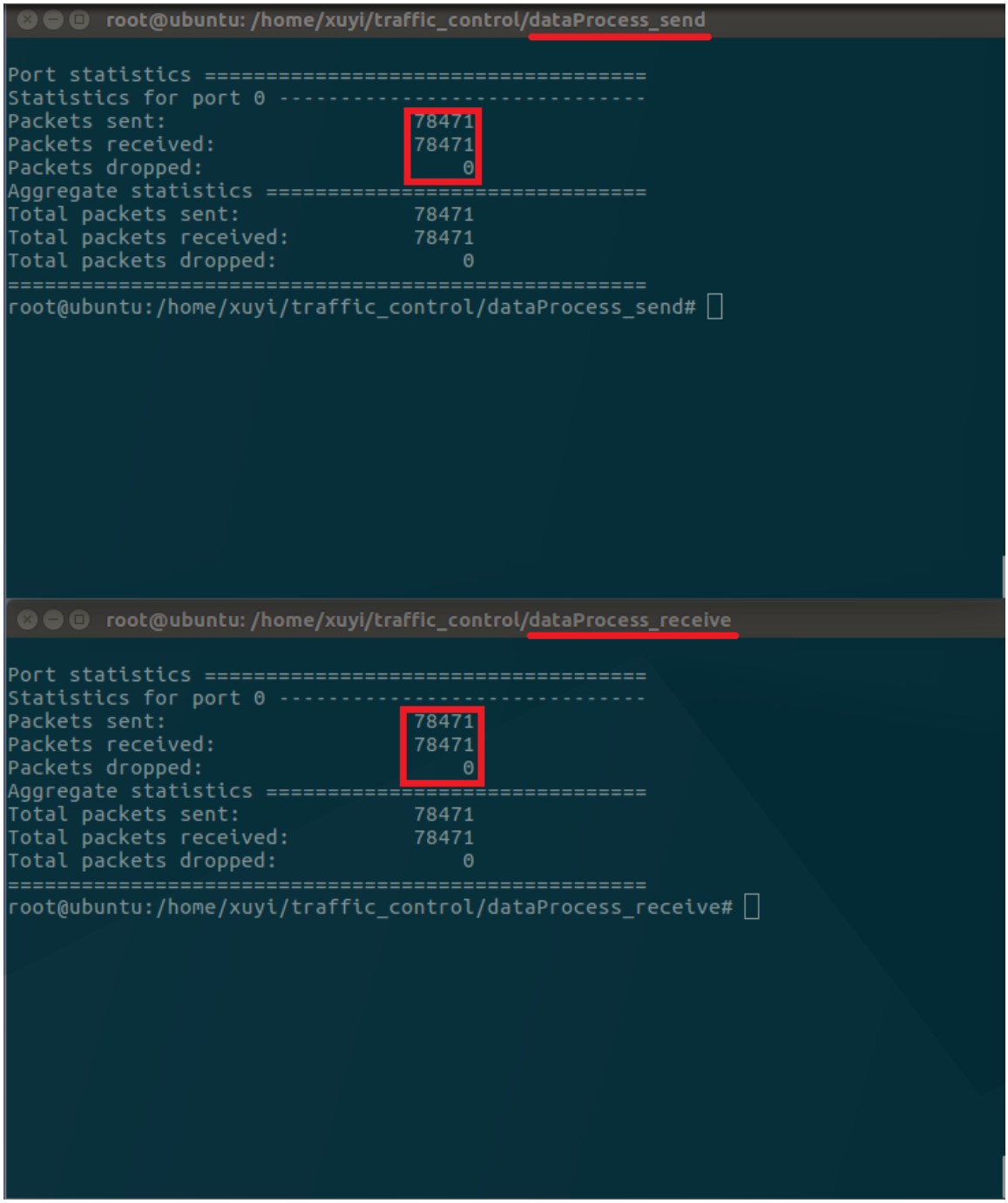

图 7: 进行流量控制后

4 makefile 相关问题

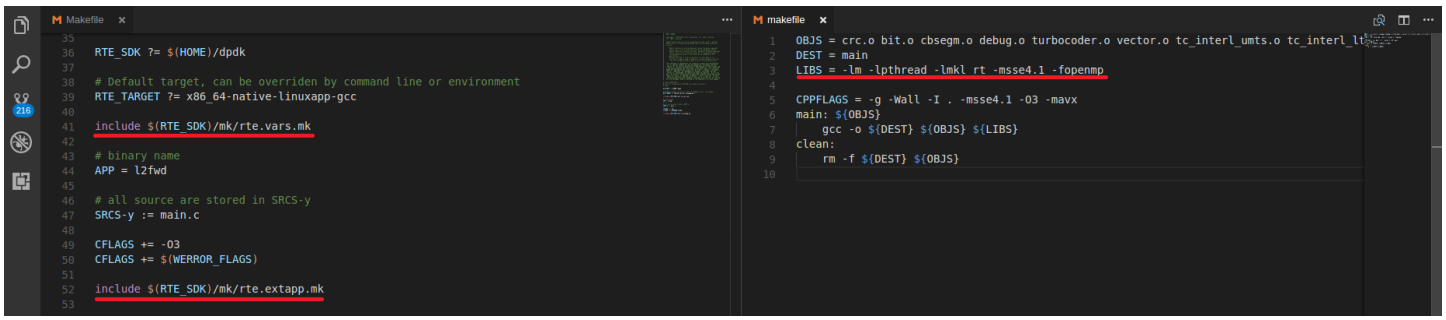

图 8: DPDK 和编码调制系统 makefile 对比

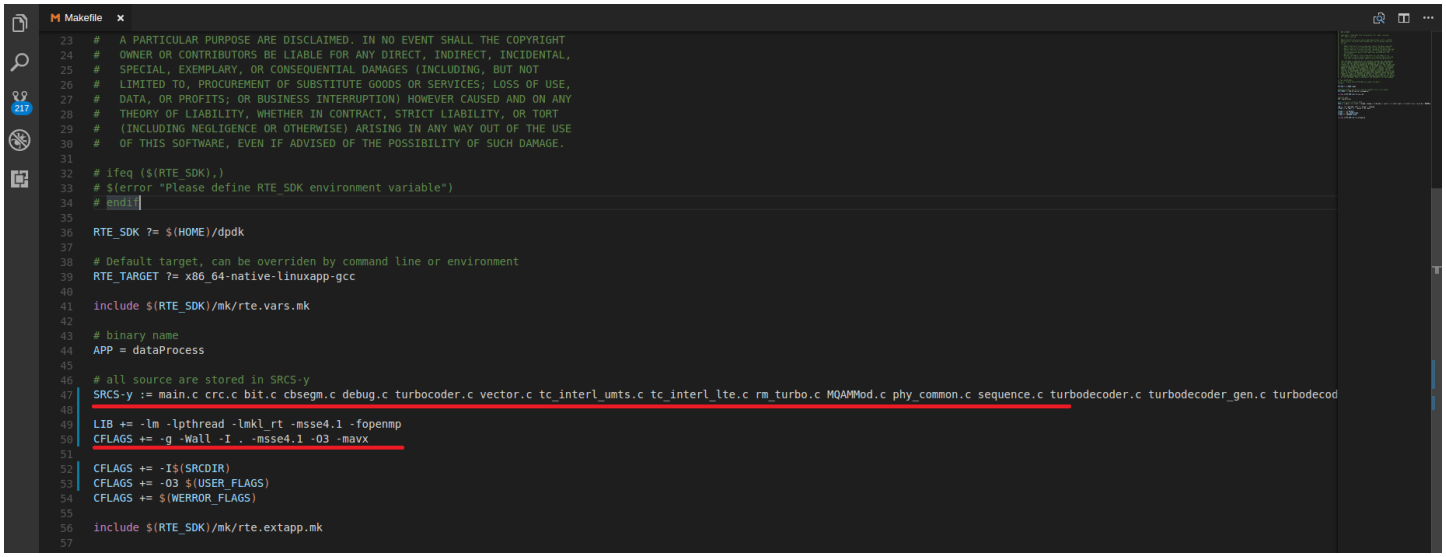

图 9: 尝试合并的 makefile

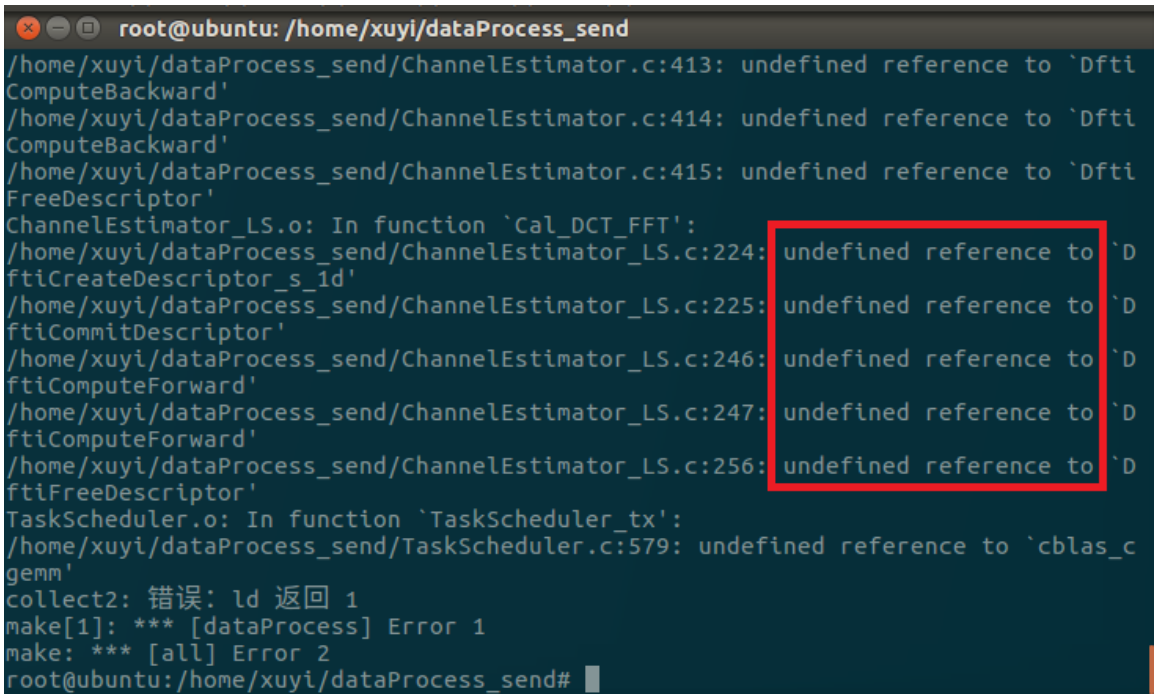

图 10: 编译报错

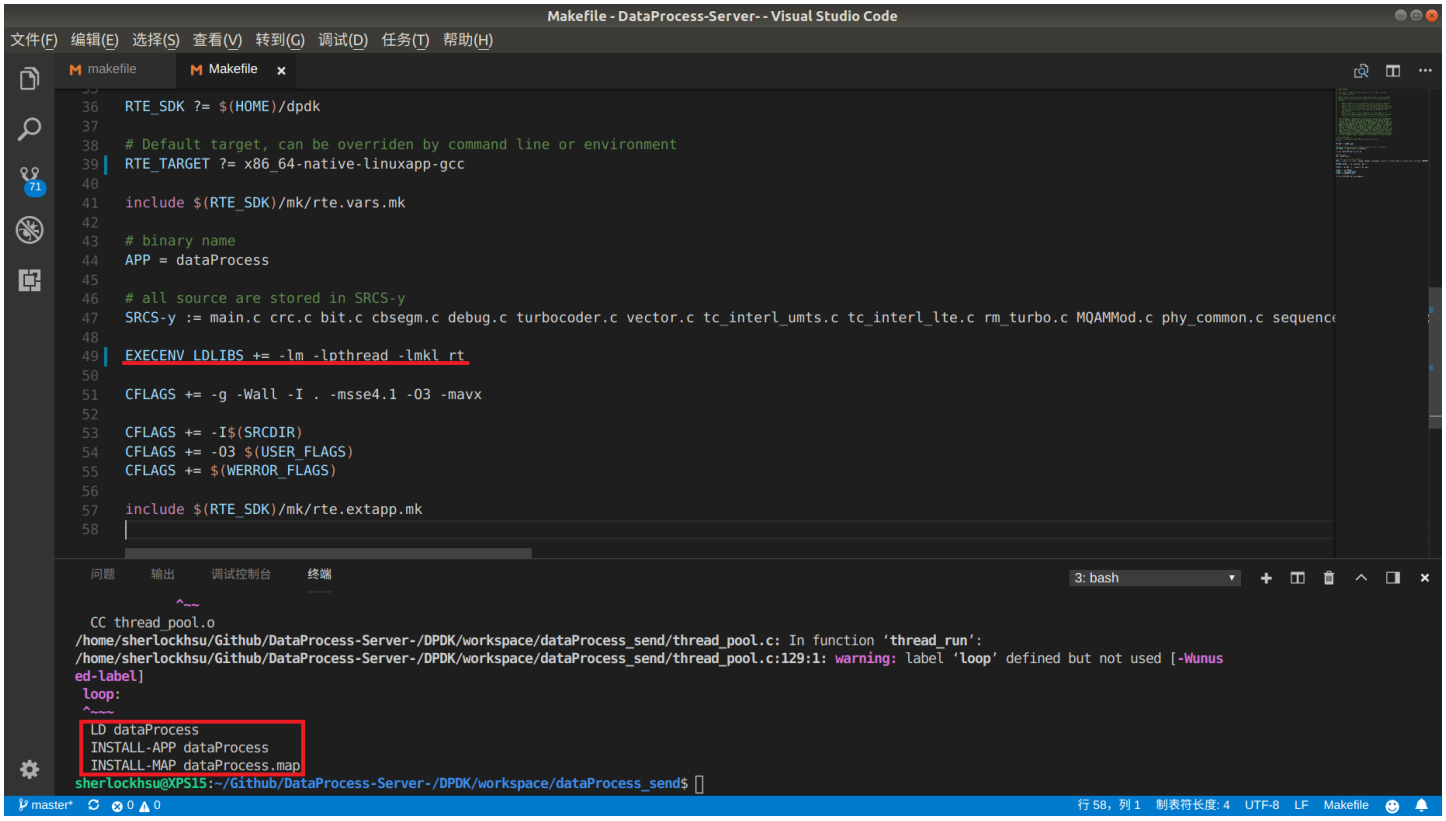

图 11: 最终解决

5 其他改进方向

1. 选择更大的 DPDK 发送页。

2. 选择更优的流量控制策略。

6 下周计划

1. 继续完成数据处理 +DPDK 系统
2. 学习 LDPC 相关内容