Embedded System Software1

과제 #1 KU\_IPC



컴퓨터공학과

201411292

이민환

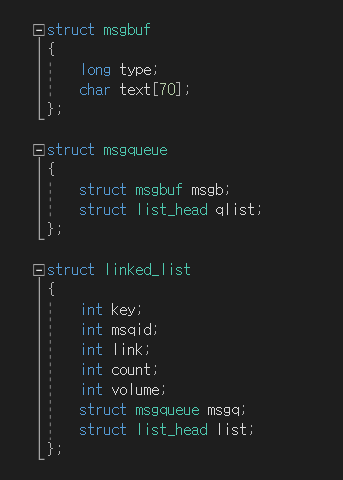
Submisson

System V message queue와 같은 방식으로 동작하는 프로그램을 만듭니다.

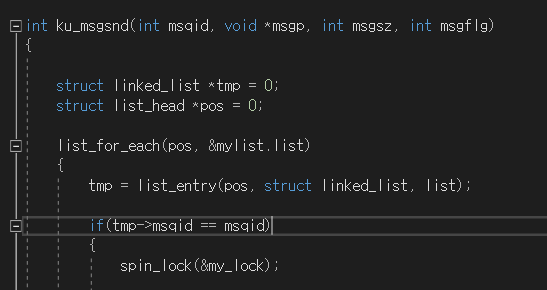
msgget()을 통하여 msgqid를 얻고 msgsnd(), msgrcv()를 통해 message를 주고받습니다. msgclose()를 하면 연결을 끊는 것이고, 연결이 모두 끊긴 queue는 삭제를 해줍니다.

Basic design

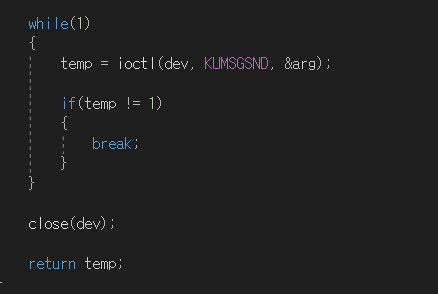
각각 다른 프로세스에서 같은 message queue에 접근을 하기 위해서 message queue를 커널 영역에 선언을 해줍니다. queue는 서로 linked list로 연결되어 있고, queue 안의 msgbuf 구조체도 linked list를 통해 연결되어 있습니다.



msgget()을 하게 되면 link가 +1이 되고, msgclose()를 하게 되면 link가 -1이 됩니다. link가 0이 되면 queue를 삭제를 합니다. 이 때, msgclose()를 하여 link가 0이 되고 삭제를 하려는 동시에, msgget()을 하여 다른 프로세스에서 msgqid를 얻게 되고 연결이 되면 안되기 때문에 spin\_lock을 걸어주어 해결을 합니다. send와 receive도 마찬가지로 동시에 같은 queue에서 동시에 일어나면 안되기 때문에, queue에 대해서는 spin\_lock을 걸어주고, 다른 queue를 사용하는 경우에는 spin\_lock을 걸어주지 않습니다.



msgflg에 따라 대기를 해야 하는 상황이 발생하기 때문에, library에서 while문 안에서 ioctl을 호출하고, return 값에 따라 while문을 빠져나가거나 무한루프를 돌면서 대기를 하기도 합니다.



msgrcv()에서 msgflg를 KU\_IPC\_NOWAIT, KU\_IPC\_NOERROR 두개 다 사용할 수 있기 때문에 NOWAIT을 1, NOERROR를 2로 define하여 OR연산을 했을 때 3이므로, 3일 때는 msgflg로 두개 다 들어왔다는 것을 알 수 있습니다.

Description for important functions

ku\_ipc\_lib

|  |  |  |
| --- | --- | --- |
| ku\_msgget() | Functionality | key와 msgflg를 구조체 안에 넣어 ioctl에 필요한 unsigned long을 구조체 주소값을 넣어주어 msgget ioctl을 호출한다. |
| Parameters | int key, int msgflg |
| Return Value | 성공하면 msqid, 실패하면 -1 |

|  |  |  |
| --- | --- | --- |
| ku\_msgclose() | Functionality | msgclose ioctl을 호출한다. |
| Parameters | int msgid |
| Return Value | 성공하면 0, 실패하면 -1 |

|  |  |  |
| --- | --- | --- |
| ku\_msgsnd() | Functionality | msgsnd에 필요한 parameters를 구조체 안에 넣어 ioctl에 필요한 unsigned long을 구조체 주소값을 넣어주어 msgsnd ioctl을 호출한다.  ioctl return 값을 통해 while 문을 빠져나가거나 대기를 위해 계속 while문을 실행한다. |
| Parameters | int msqid, void \*msqp, int msgsz, int msgflg |
| Return Value | 성공하면 0, 실패하면 -1 |

|  |  |  |
| --- | --- | --- |
| ku\_msgrcv() | Functionality | msgrcv에 필요한 parameters를 구조체 안에 넣어 ioctl에 필요한 unsigned long을 구조체 주소값을 넣어주어 msgsnd ioctl을 호출한다.  ioctl return 값을 통해 while 문을 빠져나가거나 대기를 위해 계속 while문을 실행한다. |
| Parameters | Int msqid, void \*msqp, intmsgsz, long msgtyp, int msgflg |
| Return Value | 성공하면 수신된 데이터 길이(length), 실패하면 -1 |

ku\_ipc\_c

|  |  |  |
| --- | --- | --- |
| ku\_msgget() | Functionality | parameter로 받은 key와 같은 key값을 가지는 queue를 찾아주거나, 없으면 새로 만들어준다. |
| Parameters | int key, int msgflg |
| Return Value | 성공하면 msqid, 실패하면 -1 |

|  |  |  |
| --- | --- | --- |
| ku\_msgclose() | Functionality | parameter로 받은 msqid와 같은 msqid를 가지는 queue를 찾아 프로세스와 연결을 끊어주고, 연결된 프로세스가 하나도 없게 되면 queue를 삭제한다. |
| Parameters | int msgid |
| Return Value | 성공하면 0, 실패하면 -1 |

|  |  |  |
| --- | --- | --- |
| ku\_msgsnd() | Functionality | queue에 message를 넣어준다. |
| Parameters | int msqid, void \*msqp, int msgsz, int msgflg |
| Return Value | 성공하면 0, 실패하면 -1, 대기가 필요하면 1 |

|  |  |  |
| --- | --- | --- |
| ku\_msgrcv() | Functionality | queue에 있는 message를 parameter로 받은 msqp에 넣어준다. |
| Parameters | int msqid, void \*msqp, intmsgsz, long msgtyp, int msgflg |
| Return Value | 성공하면 수신된 데이터 길이(length), 실패하면 -1, 대기가 필요하면 0 |