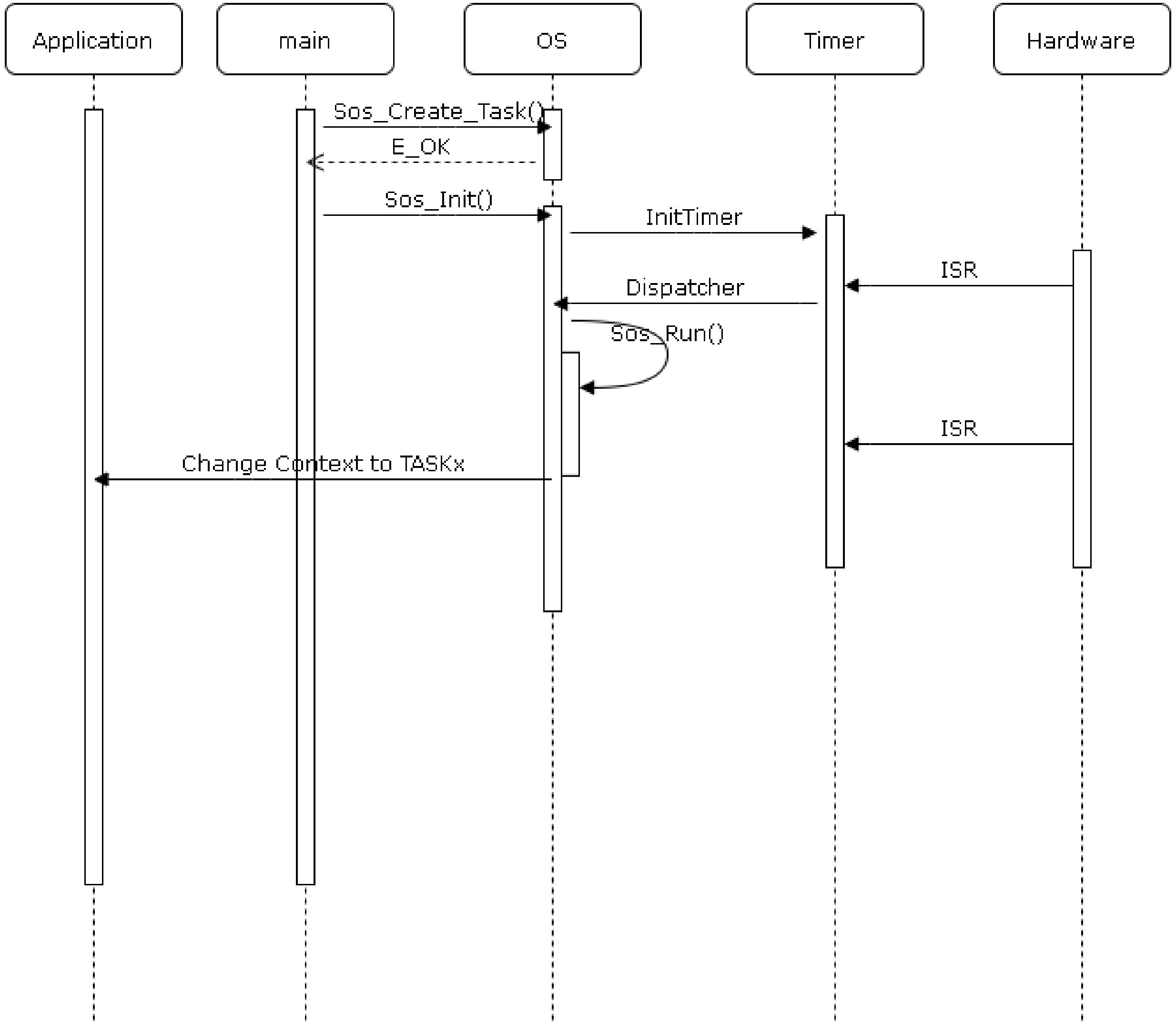
Design small OS using Time Trigger

Sequence Digram (Based on FreeRTOS)



Sos APIs

- Sos_Init()
- Sos_Run()
- Sos_Create_Task()
- Sos_Delete_Task()

Sos_Init()

```
ErrorStatus Sos_Init(void){

    Timer_Init();
    List *readyList;
    List *waitList ;

}
```

Sos_Run()

```
ErrorStatus  Sos_Run(void){

Save context ;
Evaluate waiting list ;
 Remove tasks from waiting list ;
Add tasks to ready list ;
Evaluate ready list ;
Assign task to change to its context ;
Load context of the task to be running ;

}
```

Sos_Create_Task

```
ErrorStatus  Sos_Create_Task( char* name , uint8 priority , uint8
stacksize ,void (*task)(void)){


TCB_Type * TCB ;

TCB->name=name;

TCB->priority =priority ;

TCB->stacksize =stacksize ;

TCB->task=task;

Malloc((stacksize*sizeof(Uunt8)));

AddTaskToReadyList(TCB );


}
```

Sos_Delete_Task

```
ErrorStatus  Sos_Delete_Task(void (*task)(void)){


GetTCB (Task);
RemoveTaskFromReadyList(TCB );
RemoveTaskFromWaitingList(TCB );
RemoveTaskFromEventList(TCB );
Free(TCB->stack);
Free TCB;


}
```