

12. Secondary Programs

Secondary program is executed by the robot controller concurrently and simultaneously with the primary script program. It could be used to handle the I/O signals, while the primary program moves the robot between waypoints. A secondary program could be sent to controller via primary or secondary TCP/IP socket, just like any other script program and must follow the same script syntax as regular robot programs.

Notable exception is that secondary program should not use any "physical" time. In particular, it cannot contain sleep, or move statements. Secondary program must be simple enough to be executed in a single controller step, without blocking the realtime thread.

Exceptions on secondary program do not stop execution of the primary program. Exceptions are reported in robot controller log files.

The secondary function must be defined using the keyword "sec" as follows:

```
sec secondaryProgram() :  
    set_digital_out(1,True)  
end
```

12.1. Secondary Program Limitations

Secondary programs are typically run as support programs, they are designed to be executed concurrently with the primary program using real time control loop. Entire execution time is added to single real time cycle. Due to their nature they have some limitations in using specific commands.

Motion Commands: Using any motion commands in secondary program such as `movej`, `movel` and `servoj` will not work. They will cause an error of "Runtime is too much behind". The motion commands depend on the external factor (real robot motion) and are waiting for the result of the execution, which is not possible in secondary programs.

Interpreter Mode: Sending new commands during runtime essentially goes against the design of secondary programs. This will again give you the error of runtime too much behind.

Socket: Socket communication can be used within secondary programs, but this is limited to executing simple non-blocking commands like single `socket_send`. Slow remote server or attempt to execute multiple socket commands will cause "Runtime too much behind", or prevent controller from executing real-time tasks. Socket commands should be avoided in secondary programs.

XMLRPC: Like socket, XMLRPC calls can be used in secondary programs although they need to be executed very quickly by remote server. XMLRPC calls should be avoided in secondary programs.

Sleep: Sleep, similar to motion commands is a blocking call by nature. It requires multiple real-time cycles to complete. Using `Sleep` will cause an error of "Runtime is too much behind"

Threads: Threads in secondary programs are not supported. Thread code execution is not guaranteed, and leads to undefined behaviour.