Signals in linux

Time line

P1
resumes to running state

Doing its work

Sends signal using **kill()** → Signal to be sent is captured by kernel, so that it can be delivered when P2 comes to execution

Doing its work

context switch

Other processes executes as per scheduling
Severel context switches may happen at this time

P2
resumes to running state

Pending signals are delivered to P2 by kernel → Current execution is stopped, process jumps to signal handler → Signal handler**** (for example **my_signal_handler()**) execution

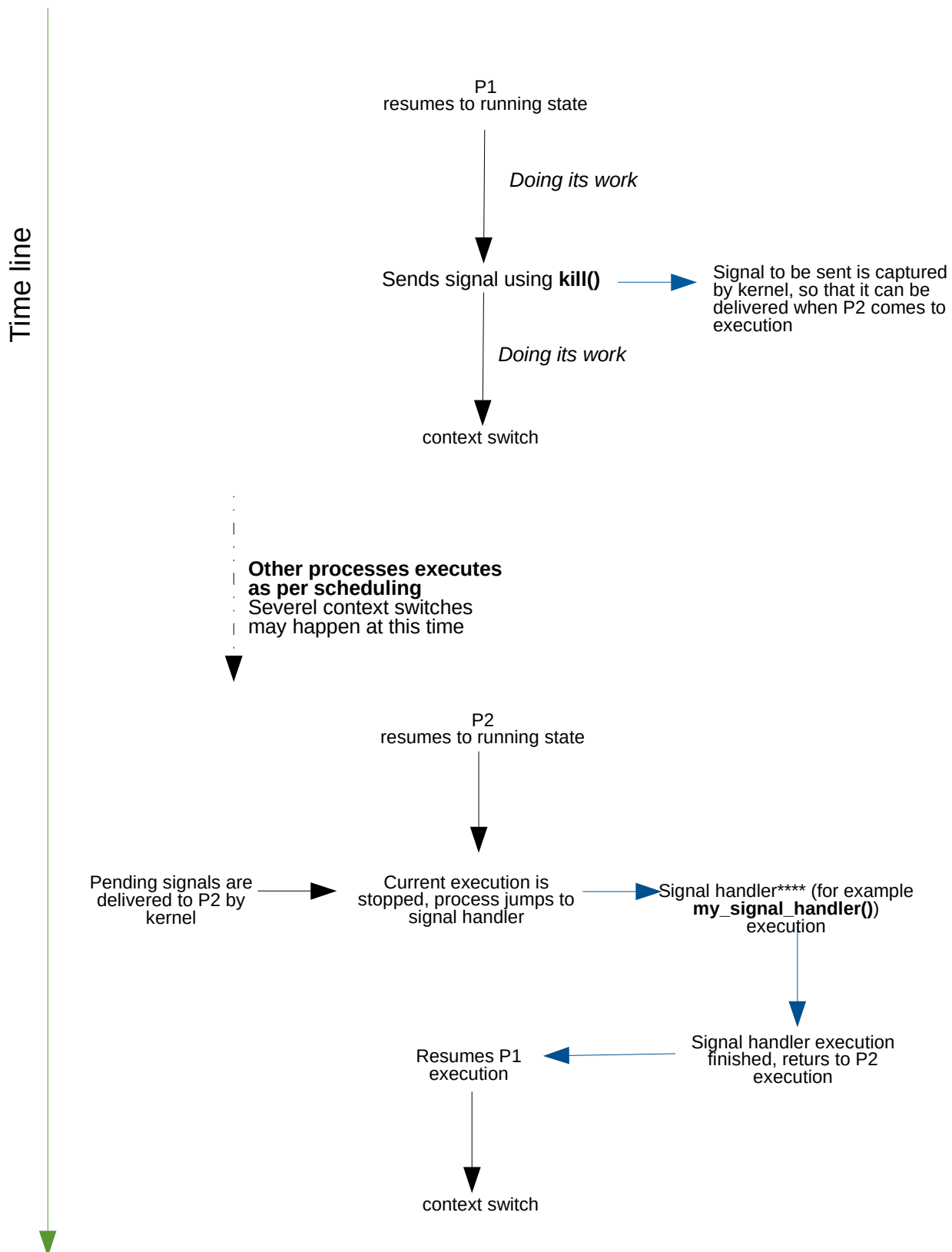Signal handler execution finished, returs to P2 execution

Resumes P1 execution

context switch

**** Signal handler has to be registered to linux using **signal()** or **sigaction()**. Signal handler is a function similar to ISR in microcontrollers.