# POU: Read_GPS

```
PROGRAM  Read_GPS
VAR
    command1  :  STRING  :=  'sudo killall gpsd' ;
    command2  :  STRING  :=  'sudo gpsd /dev/ttyAMA0 -F /var/run/gpsd.sock' ;
    command3  :  STRING  :=  'sudo killall gpsdData.py' ;
    command4  :  STRING  :=  '/home/pi/./gpsdData.py  &' ;

    Result1  :  RTS_IEC_RESULT ;
    Result2  :  RTS_IEC_RESULT ;
    Result3  :  RTS_IEC_RESULT ;
    Result4  :  RTS_IEC_RESULT ;

    xGetGPS :  BOOL := FALSE ;
    Modbus_Ready_timer :  Ton ;
    gpsdRunning :  BOOL ;
    gpsd_daemon_Ready_timer :  Ton ;
    kill_timer :  Ton ;
    kill_cmd :  BOOL := FALSE ;
END_VAR

```

```
//Attention: allow shell commands in CODESYSControl.cfg

//Program start timers
Modbus_Ready_timer ( IN := xGetGPS ,  PT := T#30S ,  Q => ,  ET => ) ;
gpsd_daemon_Ready_timer ( IN := gpsdRunning ,  PT := T#5S ,  Q => ,  ET => ) ;
kill_timer ( IN := kill_cmd ,  PT := T#5S ,  Q => ,  ET => ) ;

//After bootup, wait 30s for modbus server to run, then execute gpsd daemon
IF  Modbus_Ready_timer . q  THEN
    SysProcess . SysProcessExecuteCommand ( command1 ,  pResult  :=  ADR ( Result1 ) ) ;
    kill_cmd := TRUE ;

    //wait until existing process is killed
    IF  kill_timer . Q  THEN
        SysProcess . SysProcessExecuteCommand ( command2 ,  pResult  :=  ADR ( Result2 ) ) ;
        kill_cmd := FALSE ;
        gpsdRunning := TRUE ;
        xGetGPS  :=  FALSE ;
    END_IF

END_IF

//After gpsd daemon is running, run GPS packet decoder
IF  gpsd_daemon_Ready_timer . Q  THEN
    SysProcess . SysProcessExecuteCommand ( command3 ,  pResult  :=  ADR ( Result3 ) ) ;
    kill_cmd := TRUE ;

    //wait until existing process is killed
    IF  kill_timer . Q  THEN
        SysProcess . SysProcessExecuteCommand ( command4 ,  pResult  :=  ADR ( Result4 ) ) ;
        kill_cmd := FALSE ;
        gpsdRunning := FALSE ;
    END_IF

END_IF
```