

Go2 SDK Development Guide

Search article titles

About Go2

Remote Controller

> SDK Concepts 3

> Application Development 7

< Software Interface Services 19

Motion Services Interface V2.0

DDS Services Interface

Avoidance Services Interface

VuiClient Interface

RobotStateClient Interface

Motion Switcher Service Interface

Sports Services Interface

AI motion Services interface

Basic Services Interface

LiDAR Services Interface

Multimedia Services Interface

UWB Services Interface

Fault Services Interface

SLAM and Navigation Services Interfa

Aruco Recharge Service Interface

Odometer Services Interface

D1 Mechanical Arm Services Interface

ROS2 Services Interface

Python Services Interface

> Case reference 8

Debugging Specification

FAQ

- **Ai sport control interface:** By calling the sport_client of SDK, Go2 is sent motion commands such as speed control, position control, crouching and standing.
- **Ai sport state interface:** By subscribing to the sportmodestate message in the SDK, the position, speed, attitude and other motion states of Go2 can be obtained.

Ai sport control interface

The calling method of Ai sport control interface

The control interface adopts Api mode for users to interact with Go2 motion control module. You can call the SportClient class in the SDK to program, the example program is as follows.

```
/*
 * @file sport_client_test.cpp
 * @brief Use the class named SportClient to request sport mode
 * @date 2023-12-20
 */
#include <unitree/robot/go2/sport/sport_client.hpp>
#include <unistd.h>

//Forced close progress
void MySigintHandler(int sig)
{
    exit(0);
}

int main()
{
    signal(SIGINT, MySigintHandler);

    //Init channel
    unitree::robot::ChannelFactory::Instance()->Init();

    //Create SportClient
    unitree::robot::SportClient sport_client;

    //Set time-out period for request
    sport_client.SetTimeout(10.0f);

    //Init SportClient
    sport_client.Init();

    //Use api to balance stand
    sport_client.BalanceStand();

    sleep(3);

    //Use api to sit down
    sport_client.StandDown();

    sleep(3);

    return 0;
}
```

Introduction of Ai motion control interface

The control interface can realize the speed, position control and mode switching of Go2. The related functions of the motion control interface are listed below.

Function Name	Damp
Function Prototype	int32_t Damp()
Function Overview	Enter damping state
Parameter	None
Return Value	If the call is successful, 0 will be returned. Otherwise, relevant error codes will be returned