**摘要：** 由于最近在开发相关地图应用,所以不可必免的就会遇到GPS坐标(WGS-84)转换成国家测绘局标准要求的GCJ-02(所谓的火星坐标) 关于WGS-84与GCJ-02可以参照这篇文章,坐标转换、偏移.

由于最近在开发相关地图应用,所以不可必免的就会遇到GPS坐标(WGS-84)转换成国家测绘局标准要求的GCJ-02(所谓的火星坐标)

关于WGS-84与GCJ-02可以参照这篇文章,[坐标转换、偏移](http://bbs.amap.com/thread-18617-1-2.html).

下面的代码摘直网上,特收藏下

//World Geodetic System ==> Mars Geodetic System

//translate from https://on4wp7.codeplex.com/SourceControl/changeset/view/21483#353936

var WGS84\_to\_GCJ02 = function() {}

WGS84\_to\_GCJ02.prototype.a = 6378245.0;

WGS84\_to\_GCJ02.prototype.ee = 0.00669342162296594323;

WGS84\_to\_GCJ02.prototype.transform = function(wgLat, wgLon) {

if (this.outOfChina(wgLat, wgLon)) {

return [wgLat, wgLon];

}

dLat = this.transformLat(wgLon - 105.0, wgLat - 35.0);

dLon = this.transformLon(wgLon - 105.0, wgLat - 35.0);

radLat = wgLat / 180.0 \* Math.PI;

magic = Math.sin(radLat);

magic = 1 - this.ee \* magic \* magic;

sqrtMagic = Math.sqrt(magic);

dLat = (dLat \* 180.0) / ((this.a \* (1 - this.ee)) / (magic \* sqrtMagic) \* Math.PI);

dLon = (dLon \* 180.0) / (this.a / sqrtMagic \* Math.cos(radLat) \* Math.PI);

mgLat = wgLat + dLat;

mgLon = wgLon + dLon;

return [mgLat, mgLon];

};

WGS84\_to\_GCJ02.prototype.outOfChina = function(lat, lon) {

if (lon < 72.004 || lon > 137.8347)

return true;

if (lat < 0.8293 || lat > 55.8271)

return true;

return false;

};

WGS84\_to\_GCJ02.prototype.transformLat = function(x, y) {

var ret = -100.0 + 2.0 \* x + 3.0 \* y + 0.2 \* y \* y + 0.1 \* x \* y + 0.2 \* Math.sqrt(Math.abs(x));

ret += (20.0 \* Math.sin(6.0 \* x \* Math.PI) + 20.0 \* Math.sin(2.0 \* x \* Math.PI)) \* 2.0 / 3.0;

ret += (20.0 \* Math.sin(y \* Math.PI) + 40.0 \* Math.sin(y / 3.0 \* Math.PI)) \* 2.0 / 3.0;

ret += (160.0 \* Math.sin(y / 12.0 \* Math.PI) + 320 \* Math.sin(y \* Math.PI / 30.0)) \* 2.0 / 3.0;

return ret;

};

WGS84\_to\_GCJ02.prototype.transformLon = function(x, y) {

var ret = 300.0 + x + 2.0 \* y + 0.1 \* x \* x + 0.1 \* x \* y + 0.1 \* Math.sqrt(Math.abs(x));

ret += (20.0 \* Math.sin(6.0 \* x \* Math.PI) + 20.0 \* Math.sin(2.0 \* x \* Math.PI)) \* 2.0 / 3.0;

ret += (20.0 \* Math.sin(x \* Math.PI) + 40.0 \* Math.sin(x / 3.0 \* Math.PI)) \* 2.0 / 3.0;

ret += (150.0 \* Math.sin(x / 12.0 \* Math.PI) + 300.0 \* Math.sin(x / 30.0 \* Math.PI)) \* 2.0 / 3.0;

return ret;

};

使用例子如下:

new WGS84\_to\_GCJ02().transform(31.283814, 121.502191)

// -> [31.28181188043995, 121.50661885748906] （该坐标是上海同济大学）

最后附上一个在线转换接口,[在线转换坐标接口](http://www.zdoz.net/apiList.html)