

轮趣科技

STM32 读取镭神雷达数据 资料说明

推荐关注我们的公众号获取更新资料



版本说明:

版本	日期	内容说明
V1.0	2022/4/1	第一次发布
V1.1	2023/1/10	新增 M10P N10P

网址: www.wheeltec.net

目录

1. N10 雷达	3
1.1 接线定义	3
1.2 输出数据定义	3
1.3 输出数据案例分析	4
1.4 输出角度数据坐标定义	7
2. N10_P 雷达	8
2.1 接线定义	8
2.2 输出数据定义	8
2.3 输出数据案例分析	9
2.4 输出角度数据坐标定义	10
3. M10 雷达	11
3.1 接线定义	11
3.2 输出数据定义	11
3.3 输出角度坐标定义	12
4. M10_P 雷达	13
4.1 接线定义	13
4.2 输出数据定义	13
4.3 输出角度坐标定义	15
5. 小车避障程序说明	16
5.1 接线说明	16
5.2 雷达安装朝向	17
5.3 程序运行流程	18

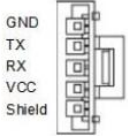
1. N10 雷达

N10 雷达能够对周围 360 度环境进行二维扫描，测试精度+3cm，角度分辨率是 0.8 度，最大量程 12m。

1.1 接线定义

4.2.2 接口定义说明

管脚	颜色	描述	最小值	典型值	最大值
GND	黑色	供电电压负极	0V	0V	0V
TX	绿色	雷达数据输出	0V	3.3V	3.5V
RX	黄色	功能控制脚	0V	3.3V	3.5V
VCC	红色	供电电压正极	4.75V	5V	5.25V
Shield	编织线	-	-	-	-



接插件型号: SMH200-05H
压线端子型号: YST200-CRT

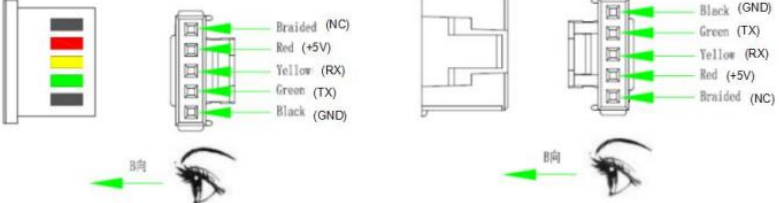


图 1-1 端口定义

1.2 输出数据定义

雷达使用串口进行通信，波特率为 230400 bps。点云输出协议如下：

表 1-1 输出数据定义

Byte_0	Byte_1	Byte_2	Byte_3	Byte_4
A5	5A	Length	Speed_H	Speed_L
Byte_5	Byte_6	Byte_7	Byte_8	Byte_9
Start_angle_H	Start_angle_L	Distance_1_H	Distance_1_L	PEAK_1
Byte_10	Byte_11	Byte_12
Distance_2_H	Distance_2_L	PEAK_2
Byte_55	Byte_56	Byte_57		
Stop_angle_H	Stop_angle_L	CRC		

1. Byte_0 - Byte_1: 为帧头，固定值。
2. Byte_2 : 整个数据帧的长度，从帧头到校验位。

3. Byte_3- Byte_4 : 转速信息, 一个码盘的时间, 高位在前, 单位为 us。

例如: Speed_H=0x10, Speed_L=0x46,即 0x1046—4166us。

码盘转一圈时间: $4166\text{us} \times 24 = 100\text{ms}$,即转速 10Hz。

4. Byte_5- Byte_6: 一帧数据的起始角度, 高位在前, 是实际角度的 100 倍。

例如: Start_angle_H=0x42, Start_angle_L=0x08, 即 0x4208—16904—169.04 度。

5. Byte_7- Byte_54: 点云数据: 每个数据包括 2 字节距离 1 字节强度信息, 高位在前, 距离单位 mm 例如: Distance_1_H=0x00, Distance_1_L=0x64, PEAK_1=0x64, 表示距离 $0x64 = 100\text{mm}$, 强度 100。

6. Byte_55- Byte_56: 结束角度。与起始角度一样算法。

7. Byte_57: 从 Byte_0 到 Byte_56 数据和校验值。

$\text{CRC} = \text{byte0} + \text{byte1} + \dots + \text{byte56}$

1.3 输出数据案例分析

N10 雷达的数据角度分辨率是 0.8 度, 所以雷达转一圈那么会输出 450 个数据, 但雷达一包数据包输出的点是 16 个点, 按此可知, 雷达一圈能输出差不多 29 个数据包, 输出一包数据包的角度范围大概是 12 度。雷达接上串口后会自动以波特率 230400 发送数据包, 不需要发送其他命令。数据包之间间隔一个点的角度, 也就是角度分辨率 0.8 度。以下是使用串口助手接收到的数据, 可以看出 [A5 5A 3A 10 5D 17 0E 01 55 3F 01 44 2C 01 C6 2C 01 D6 0F 01 C6 32 01 C6 6F 01 C6 8E 01 C6 88 01 55 41 01 C6 4C 01 C6 57 01 C6 9B 01 C6 93 01 C6 85 01 C6 76 01 C6 68 1B BA 8E]是其中的一个数据包。我们对其进行解析。



图 1-2 串口助手接信息

表 1-2 数据帧说明

数据内容	帧头 1	帧头 2	数据帧的长度	转速(单位 us)		起始角度		点云数据 1		
说明	固定值 0XA5	固定值 0X5A	单位: 字节	高 8 位	低 8 位	高 8 位	低 8 位	距离信息 高 8 位	距离信息 低 8 位	强度信息
占用字节	1	1	1	1	1	1	1	1	1	1
序号	1	2	3	4	5	6	7	8	9	10
数据内容	点云数据 2			点云数据 16			结束角度		校验和
说明	距离信息 高 8 位	距离信息 低 8 位	强度信息	距离信息 高 8 位	距离信息 低 8 位	强度信息	高 8 位	低 8 位	前面数据相加
占用字节	1	1	1	1	1	1	1	1	1
序号	11	12	13	53	54	55	56	57	58

① 帧头和数据帧长度

帧头一共两个字节，固定值是 0XA5 和 0X5A。

数据帧长度也是固定的，值为 0X3A，单位为字节，即 58 字节。

② 转速

这里输出一个齿的时间。转一圈是一个码盘的时间，一个码盘有 24 个齿。

转速一共两个字节，单位是 μs 。分析捕获到的转速信息[10 5D]，即 0X105D = 4189 μs ，那么转一圈就是 $4189 \times 24 = 100.536\text{ms}$ ，转速约为 10Hz。

注：十六进制换算成十进制可使用 windows 自带计算器，切换到程序员模式，在 HEX 栏输入十六进制数据，在 DEC 栏可得到十进制数据。如需手动计算请上网查找相关方法。

③ 起始角度和结束角度

角度信息是放大了 100 倍的信息，如捕获到的起始角度信息[17 0E]，即 $0X170E = 5902 = 59.02$ 度；结束角度信息[1B BA]，即 $0X1BBA = 7098 = 70.98$ 度。

④ 点云数据

一包数据包会输出 16 个点的数据，每个点的数据包含距离和强度信息。比如捕获到的第一个点云数据[01 55 3F]，可以知道距离是 $0X0155 = 341\text{mm}$ ，强度为 $0X3F = 63$ 。

⑤ 校验和

校验和是前面所有数据相加的结果，只要 8 位数据。

1.4 输出角度数据坐标定义

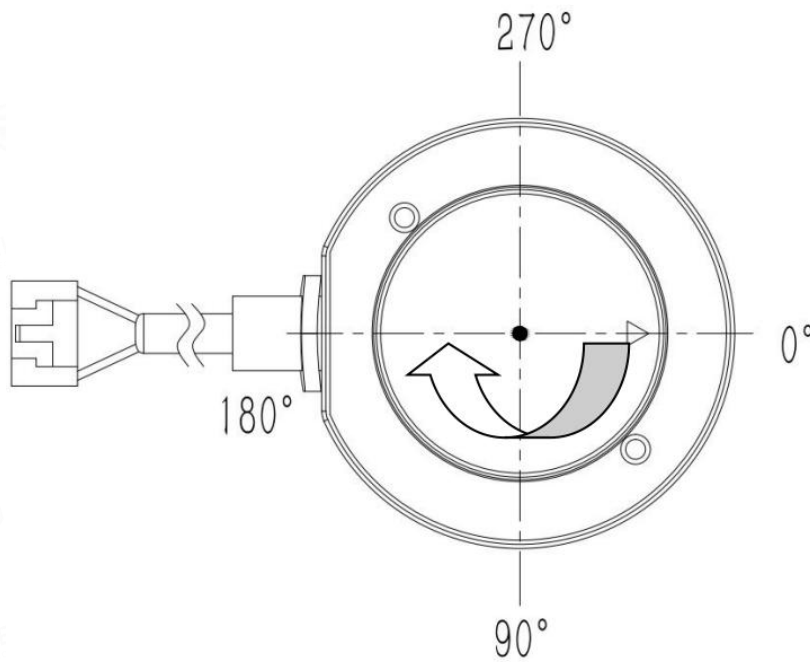


图 1-3 N10 雷达坐标系定义

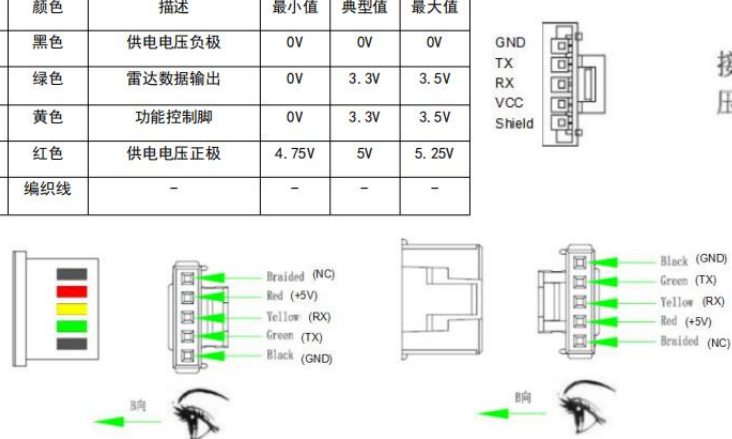
2. N10_P 雷达

N10P 是一款室内外通用的 TOF 测距雷达，测试精度+3cm，角度分辨率是 0.4 度，最大量程 12m。

2.1 接线定义

4.2.2 接口定义说明

管脚	颜色	描述	最小值	典型值	最大值
GND	黑色	供电电压负极	0V	0V	0V
TX	绿色	雷达数据输出	0V	3.3V	3.5V
RX	黄色	功能控制脚	0V	3.3V	3.5V
VCC	红色	供电电压正极	4.75V	5V	5.25V
Shield	编织线	-	-	-	-



接插件型号: SMH200-05H
压线端子型号: YST200-CRT

图 2-1 端口定义

2.2 输出数据定义

雷达使用串口进行通信，波特率为 460800 bps。点云输出的数据协议为 108 个字节，包含了角度，转速，距离和强度信息，点云输出协议如下：

表 2-1 输出数据定义

Byte_0	Byte_1	Byte_2	Byte_3	Byte_4
A5	5A	Length	Speed_H	Speed_L
Byte_5	Byte_6	Byte_7	Byte_8	Byte_9
Start_angle_H	Start_angle_L	Dist_1_1_H	Dist_1_1_L	PEAK_1_1
Byte_10	Byte_11	Byte_12	Byte_13	Byte_14
Dist_1_2_H	Dist_1_2_L	PEAK_1_2	Dist_2_1_H	Dist_2_1_L
Byte_15	Byte_16	Byte_17	Byte_18
PEAK_2_1	Dist_2_2_H	Dist_2_2_L	PEAK_2_2
Byte_103	Byte_104	Byte_105	Byte_106	Byte_107
预留位	预留位	Stop_angle_H	Stop_angle_L	CRC

注：L 表示数据低位，H 表示数据高位。

1. Byte_0 - Byte_1：为帧头，固定值。

2. Byte_2：整个数据帧的长度，从帧头到校验位。

3. Byte_3- Byte_4：转速信息，一个码盘的时间，高位在前，单位为 us。

例如：Speed_H=0x10, Speed_L=0x46,即 0x1046—4166us。

码盘转一圈时间：4166us*24=100ms,即转速 10Hz。

4. Byte_5- Byte_6：一帧数据的起始角度，高位在前，是实际角度的 100 倍。

例如：Start_angle_H=0x42, Start_angle_L=0x08,即 0x4208—16904—169.04 度。

5. Byte_7- Byte_102：点云数据：每个数据包括 2 字节距离 1 字节强度信息，高位在前，距离单位 mm。例如：Dist_1_1_H=0x00, Dist_1_1_L=0x64, PEAK_1_1=0x64,表示距离 0x64=100mm,强度 100；Dist_1_1 与 Dist_1_2 对应同一个角度。

6. Byte_103- Byte_104：预留位。

7. Byte_105- Byte_106：结束角度。与起始角度一样算法。

8. Byte_107：从 Byte_0 到 Byte_106 数据和校验值。

$CRC = byte0 + byte1 + \dots + byte106$

2.3 输出数据案例分析

N10P 雷达的数据角度分辨率是 0.4 度，所以雷达转一圈那么会输出 900 个数据，N10P 雷达输出一帧数据是 32 个点，按此可知，雷达一圈能输出差不多 29 个数据包，输出一包数据包的角度范围大概是 12 度。雷达接上串口后会自动以波特率 460800 发送数据包，不需要发送其他命令。数据包之间间隔 0.4 度。N10P 雷达数据的解析方式与 N10 雷达数据的解析方式相同，请参阅 [1.3 输出数据案例分析](#) 的内容。

2.4 输出角度数据坐标定义

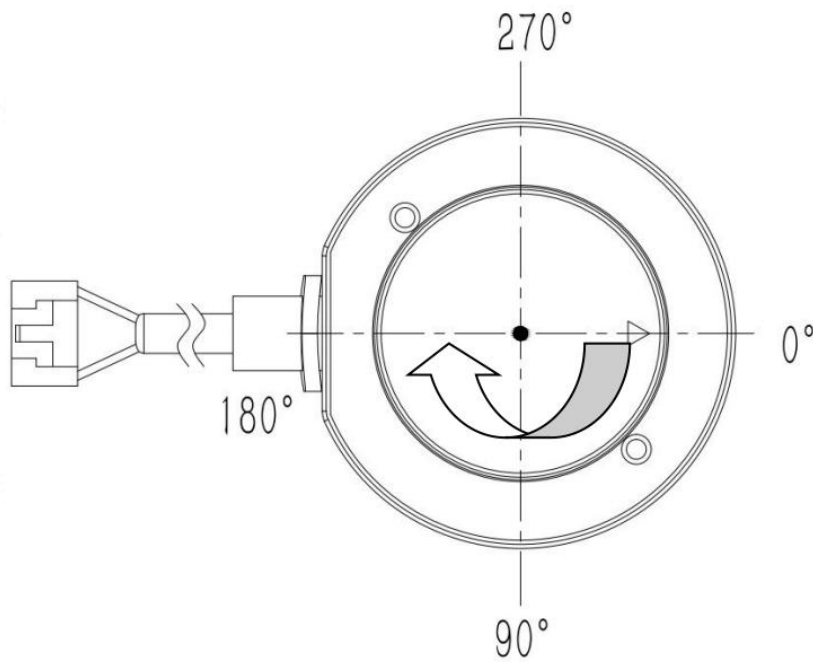


图 2-2 N10P 雷达坐标系定义

3. M10 雷达

3.1 接线定义

M10 雷达使用 6P 端子，接线定义如下：

表 1.2 雷达输出接口定义

管脚	描述	典型值	范围	备注
VCC	供电电压正极	5V	4.75~5.25	纹波 80mV 以内
GND	供电电压负极	0V	0V	
RX	系统串口输入		TTL	数据流：外设→雷达
TX	系统串口输出		TTL	数据流：雷达→外设
PPS	GPS 秒信号			
REC	GPS 经纬度时分秒			

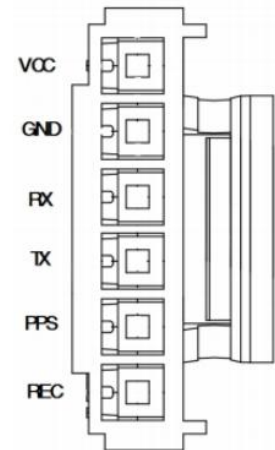


图 3-1 M10 端口定义

3.2 输出数据定义

M10 雷达使用串口进行通信，波特率为 460800 bps，一共输出 92 个字节，协议定义如下：

表 3-1 M10 雷达输出协议

Byte_0	Byte_1	Byte_2	Byte_3	Byte_4
A5	5A	angle_CodedDisc	angle_CodedDisc	Speed
Byte_5	Byte_6	Byte_7	Byte_8	Byte_9
Speed	Distance_1	Distance_1	Distance_2	Distance_2
Byte_10	Byte_88	Byte_89
Distance_3	Distance_42	Distance_42
Byte_90	Byte_91			
FA	FB			

1. A5 5A 为帧头

2. 帧头后面就是角度和转速参数，从第 6 个字节开始，就是 42 个点的距

离参数。

3. 0xFA 0xFB 为帧尾

4. 数据字节说明

angle_CodedDisc: 角度参数，一共两个字节,高位在前，低位在后，此角度为数据包结束的角度值。例如：0x8C 0xA0 即十进制 36000 表示角度为 360 度，也就是 0 度；

speed: 转速参数，一共两个字节,高位在前，低位在后，表示雷达从一个齿转到下一个齿所需要的时间计数值，它和转速的计算公式为：转速 = $2500000/\text{speed}$ 。例如：0x10 0x68 即十进制 4200 则转速为每分钟 595.239 转，也就是 10HZ；

Distance: 距离参数，一共两个字节,高位在前，低位在后，一帧一共输出 42 个点的数据，单位是毫米，角度分辨率是 0.22 度。例如：0x13 0x88 即十进制 5000 那么该角度对应的距离值就是 5 米。

3.3 输出角度坐标定义

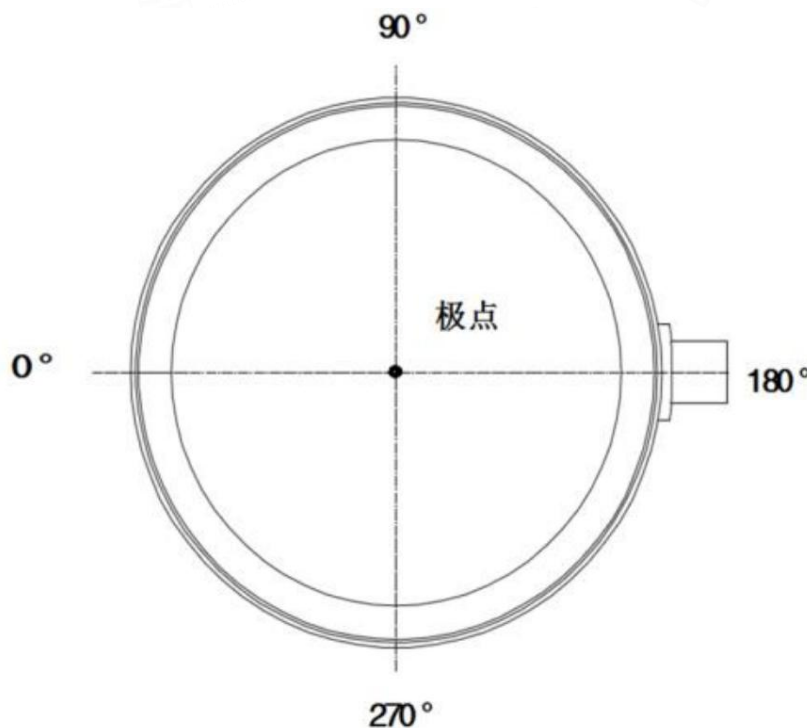


图 3-2 M10 雷达坐标定义

4. M10_P 雷达

4.1 接线定义

M10P 雷达使用 6P 端子，接线定义如下：

表 1.2 雷达输出接口定义

管脚	描述	典型值	范围	备注
VCC	供电电压正极	5V	4.75~5.25	纹波 80mV 以内
GND	供电电压负极	0V	0V	
RX	系统串口输入		TTL	数据流：外设→雷达
TX	系统串口输出		TTL	数据流：雷达→外设
PPS	GPS 秒信号			
REC	GPS 经纬度时分秒			

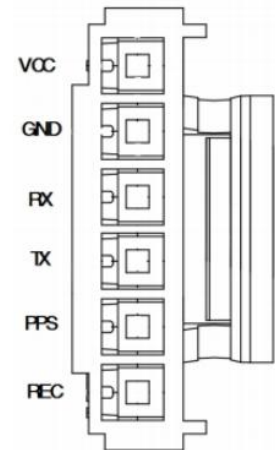


图 4-1 M10 端口定义

4.2 输出数据定义

M10P 雷达使用串口进行通信，波特率为 512000 bps，一共输出 160 个字节，协议定义如下：

表 4-1 M10P 雷达输出协议

Byte_0	Byte_1	Byte_2	Byte_3	Byte_4
A5	5A	Len_H	Len_L	Angle_H
Byte_5	Byte_6	Byte_7	Byte_8	Byte_9
Angle_L	Speed_H	Speed_L	Distance_1_H	Distance_1_L
Byte_10	Byte_11	Byte_146	Byte_147
Distance_2_H	Distance_2_L	Distance_70_H	Distance_70_L
Byte_148	Byte_157	Byte_158	Byte_159
TIME	TIME	TIME	0xFA	0xFB

注：L 表示数据低位，H 表示数据高位。

1. Byte_0 - Byte_1: 为帧头, 固定为 A5 5A。
2. Byte_2- Byte_3 : Len_H/ Len_L 数据帧长度, 从第一个字节到最后一个字节。
3. Byte_4- Byte_5 : Angle_H/ Angle_L 当前数据段开始角度, 例如: 0x8C 0xA0 即十进制 36000 表示角度为 360 度, 也就是 0 度。

例如: Speed_H=0x10, Speed_L=0x46,即 0x1046—4166us。

码盘转一圈时间: 4166us*24=100ms,即转速 10Hz。

4. Byte_6- Byte_7 : Speed_H/ Speed_L 当前电机转速, 表示雷达从一个齿转到下一个齿所需要的时间计数值, 它和转速的计算公式为: 转速=2500000/speed。
例如: 0xD 0x90 即十进制 3472 , 则转速为每分钟 720 转, 也就是 12HZ;

5. Byte_8- Byte_147 : Distance_x_H/ Distance_x_L 距离值, 单位是毫米,且高位字节的最高位表示是否为高反。例如: Distance_x_H=0x13, Distance_x_L=0x88 , 高字节 0x13 的最高位为 0 表示不是高反, 距离值对应十进制 5000, 就是 5 米。

例如: Distance_x_H=0x93, Distance_x_L=0x88 , 高字节 0x93 的最高位为 1 表示当前 M10P 这个点为高反, 距离值要清除最高位, 清除高位后对应十进制 5000, 距离值就是 5 米。

如果距离参数为 0xFFFF, 则表示此数据点为无效的点, 数据帧中的点云个数需要减一, 此时实际点云个数变成 69 个点 (正常点云个数为 70 个)。同理如果现出两个 0xffff, 则实际点云数只有 68 个。

每两个点云之间的角度差计算方法: 15 除以实际收到的点云个数 (m) 。
则数据帧每个数据点的角度为:

第 N 个点的水平角度为 $= \text{angle_CodedDisc} + 15 / m * N$ (N = 0,1 ... m-1)

6. Byte:148- Byte_157: GPS 时间信息, 暂时保留无作用 (需要硬件支持)。
7. Byte:158- Byte_159: 帧尾, 固定为 0XFA 0XFB。

4.3 输出角度坐标定义

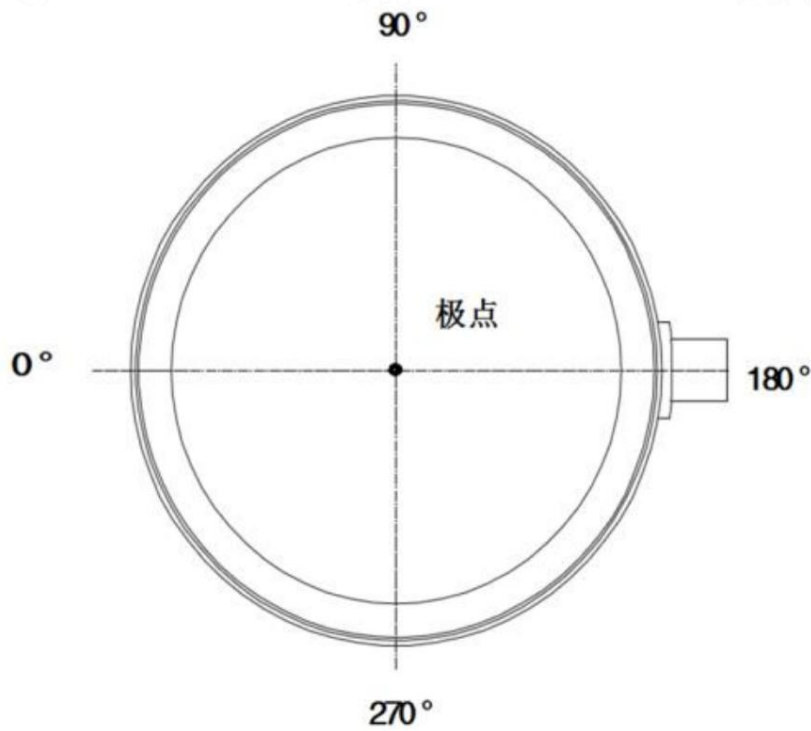


图 4-2 M10P 雷达极坐标定义

5. 小车避障程序说明

以下说明以 N10 雷达为例进行说明，程序中的 N10、M10 雷达数据处理类似，避障算法也是类似的。小车是 F4 麦轮小车，使用串口 5 读取 N10 雷达传回来的数据，因为 N10 雷达的角度分辨率是 0.8 度，不重复的点有 450 个，为了简化，将一帧数据包传回来的 16 个点做平均处理，作为一个点储存。

5.1 接线说明

小车与雷达的接线如下(N10 为例):

表 5-1 接线说明

小车	雷达
GND	GND
PD2(串口 5 的 RX)	TX
不接	RX
5V	VCC
GND	Shield

小车各对应引脚图示:



图 5-1 雷达安装接线图示

5.2 雷达安装朝向

N10 雷达安装位置：雷达的 0 度对应小车的前进的方向。



图 5-2 N10 雷达安装方向

M10 雷达安装位置：雷达的 0 度对应小车的前进的方向。



图 5-3 M10 雷达安装方向

5.3 程序运行流程

程序框架简述：雷达在上电的时候会自动传回数据包，不需要进行额外的设置，此时只需要在串口中断中进行接收即可。雷达传回的数据在串口 5 中断进行接收，并且在串口 5 中断中做一定的简化。一帧数据有 16 个点的位置信息，包括角度和距离。每次传回一帧数据后，做平均简化，然后储存在一个 50 个变量的结构体数组中，此后，每次收到新数据，不断刷新这个数组，作为避障的判断。因为小车是前进着的避障，所以这里设置了以 0 度为中心，左右两边各 50 度的范围内的避障，即前方范围 100 度内的避障，这个时候只需要前方 100 度范围内的数据，其他数据舍弃不要。在 `Balance_task` 任务中，需要做避障的判断和运动处理。假设小车有一个前进的速度，小车避障的时候设置避障距离是 220mm，在避障算法中，遍历储存的 50 个数据，找出所有距离是小于 220mm 的点，并且判断前方障碍物是位于左边还是位于右边，小车此时转动 z 轴方向，向相反的方向转向，直至前方避障范围内没有需要避障的点。