**1.金线检测：**

class DefectTypeBase

{

public:

virtual int defect\_type\_and\_height(*std*::*string* img\_path, *std*::*string* cali\_path, *std*::*string* depth\_path, *std*::*string* model\_path,int thresh1,int thresh2,int rows,int cols)=0; //得到图片的缺陷类型

virtual *std*::*vector*<int> get\_defect\_type() const = 0;//总共18根线，2块芯片

virtual *std*::*vector*<int> get\_parallel\_type()const = 0;//

virtual *std*::*vector*<float> get\_wire\_height()const = 0;//18根线的高度

};

1. virtual *std*::*vector*<int> get\_defect\_type() const = 0;//总共18根线，2块芯片

金线的排列顺序：只排到了11，总共应该是18根线的缺陷类型。

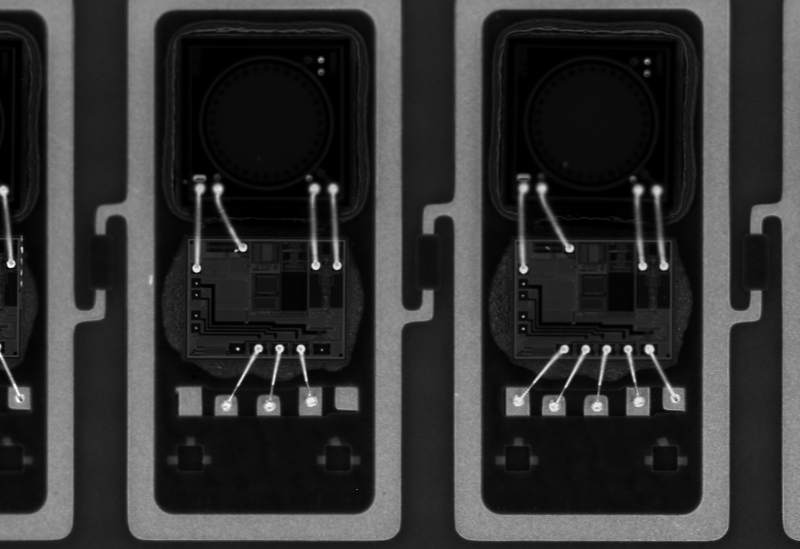
返回值是0：无线

返回值是1：堕线

返回值是2：断线/脚起

返回值是3：线偏

返回值是4：正常金线



11

10

9

8

7

6

5

4

3

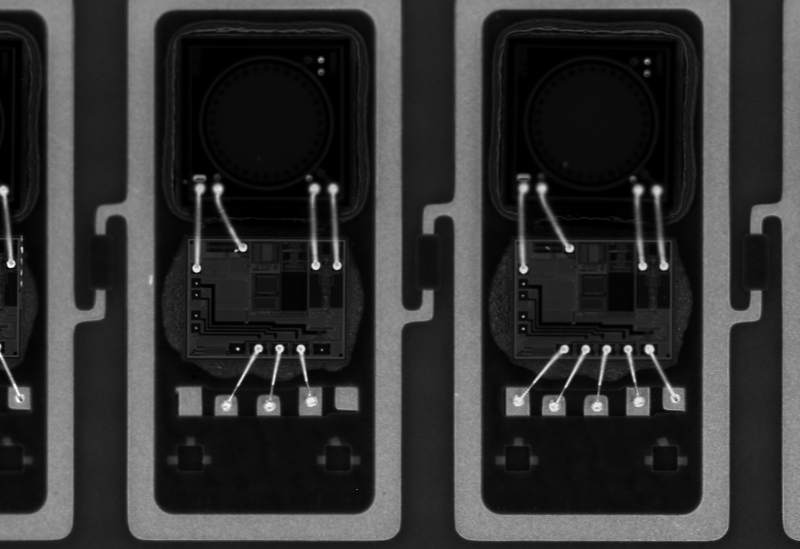
2

1

1. virtual *std*::*vector*<int> get\_parallel\_type()const = 0;//判断2根线是否是并线，总共有14个值，如下图只画了左边部分7个地方的并线

返回值0：并线

返回值1：无并线



1. virtual *std*::*vector*<float> get\_wire\_height()const = 0;//18根线的高度

返回值： 18根线的高度。

**2.球检测**

class DefectDetectorBase

{

public:

virtual *std*::*vector*<*std*::*vector*<int>> defect\_type(*std*::*string* img\_path, *std*::*vector*<int> thresh, *std*::*string* model\_path)=0;

};

返回值：总共108个球的缺陷类型，是一个6\*18的vector，意思是每个图包括6块芯片，每块芯片有18个球，如下图芯片的排列方式和芯片中球的排列方式已经注明。

enum DefectType {

DT\_INIT = 0b00000000,

DT\_NORMAL = 0b00000001,

DT\_SMALL = 0b00000010,

DT\_BIG = 0b00000100,

DT\_TAIL = 0b00001000,

DT\_NOT\_ROUND = 0b00010000,

DT\_DEVIATION = 0b00100000,

DT\_MISSING = 0b01000000

};

返回值是1：正常球

返回值是2：小球

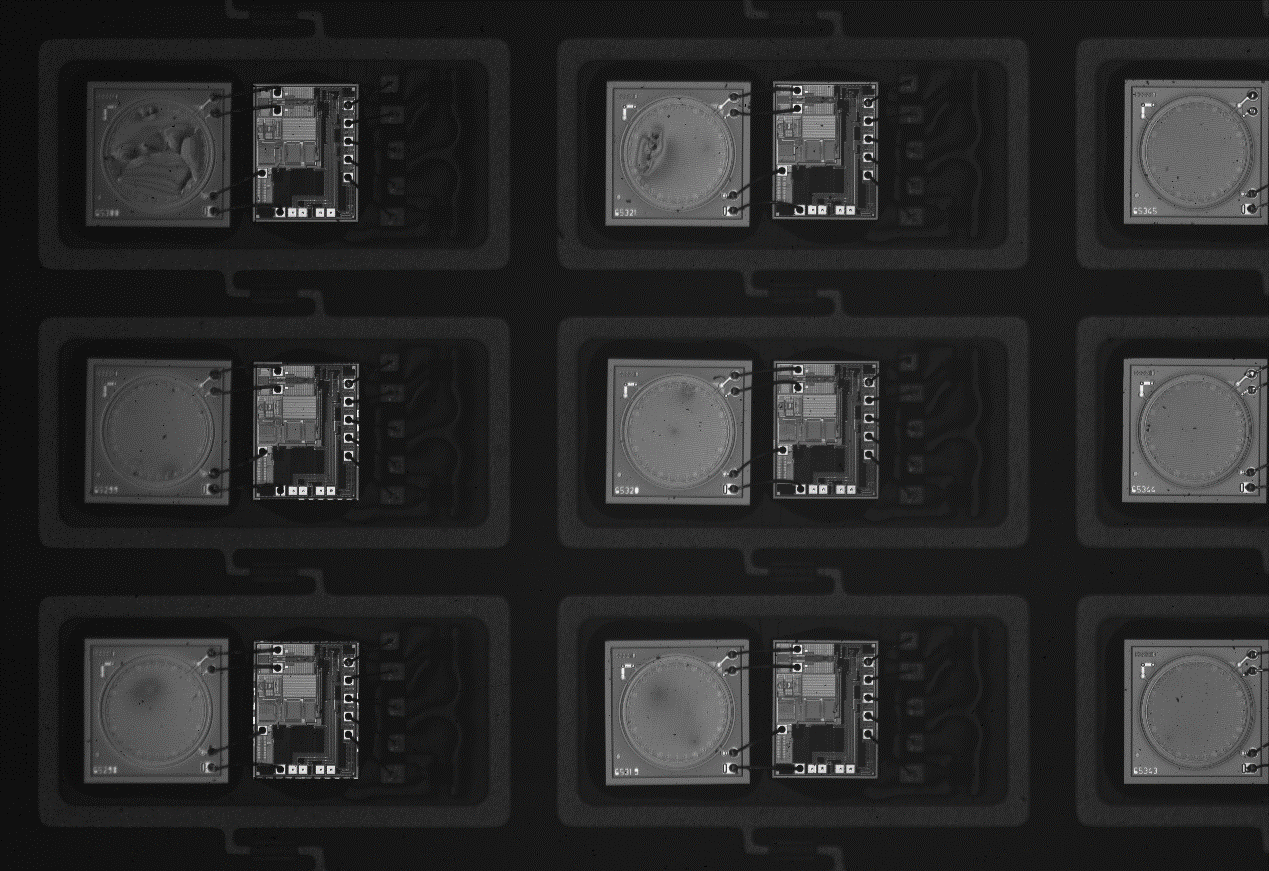
返回值是4：大球

返回值是8：线尾球

返回值是16：球缺

返回值是32：球偏

返回值是64：无球

****

芯片6

芯片5

芯片4

芯片3

芯片2

芯片1

14~184

1~4