

广东省分析测试协会团体标准
《表面化学分析 扫描探针显微镜数据存储
格式》
编制说明

《表面化学分析 扫描探针显微镜数据存储格式》编制组

2023年12月

目录

1 项目背景.....	1
1.1 本标准制定的目的与意义.....	1
1.2 本标准与国内外标准关系的说明.....	2
2 标准起草工作简况.....	3
2.1 任务来源.....	3
2.2 起草单位、起草人.....	3
2.3 主要工作过程.....	3
3 标准编制原则和确定标准主要内容的论据.....	5
3.1 标准的编制原则.....	5
3.2 确定标准主要内容的依据.....	5
4 方法验证及格式试验.....	7
4.1 基本情况.....	7
4.2 验证情况-1：广州中源仪器技术有限公司.....	8
4.3 验证情况-2：中山大学物理学院.....	17
4.4 验证情况-3：中山大学分析测试中心.....	19
4.5 验证情况-4：暨南大学理工学院.....	21
4.6 验证情况-5：华南理工大学分析测试中心.....	23
5 采用国际标准和国外先进标准情况及水平对比.....	25
6 与现行法律、法规、政策及相关标准协调性.....	26
7 公开标准涉及专利的信息.....	26
8 重大意见分歧的处理依据和结果.....	27
9 预期的社会经济效益及贯彻实施标准的要求、措施等建议.....	27
10 其他应当说明的事项.....	27

1 项目背景

项目背景，包括目的与意义、产业现状、国内外现行相关法律、法规和标准情况等。

1.1 本标准制定的目的与意义

随着扫描探针显微术（SPM）发展和应用的日益广泛。在数据格式方面，由于所用硬件平台的不同等原因，各制造商所生产的 SPM 大都使用专用的数据格式。这些数据格式大多互不兼容，给数据后续的传输、交换、共享、以及数据存储、分析处理和格式转换带来了不便。SPM 数据格式的标准化是大势所趋。

2011 年，ISO 发布了标准 ISO 28600（该标准已于 2018 年转化为国家标准 GB/T 36052-2018）。这是一个基于文本的、SPM 数据传送的格式标准。但 ISO 28600 并不是一套适合用于对 SPM 数据进行高效存储和处理的理想方案，这是因为该标准中所有数据都基于 ASCII 文本，与同信息量的基于二进制的文件相比，占用的存储空间相对较大，因此不太适合于 SPM 海量数据的存储；其数据结构不支持随机读写，数据分析处理时效率不高。

对于 SPM 数据的存储和分析处理，目前国内外还没开放的、高效的数据存储格式的方案或标准。国内外的各 SPM 制造商所生产的 SPM 仪器都使用专用和非开放的数据格式进行存储，这些 SPM 数据的格式互不兼容，大都各自开发了 SPM 数据处理和格式转化的专用软件。这导致 SPM 数据的格式转换和分析处理的软件共享仍然困难重重，基本只能以企业或组织间的商业合作为主要实现方式。近年来国际上出现了一些优秀的、具有开放性的 SPM 数据分析处理的软件平台，如著名的国际开源项目 GXSM(Gnome X Scanning Microscopy)，目前已能实现对国际上 20 多种 SPM 专用数据格式的识别和分析处理。但是，由于我国在 SPM 仪器制造领域目前还没有具备国际竞争力的仪器制造企业和国际高端的 SPM 仪器产品，在这一领域的国际合作和市场竞争中，我国仍处在非常不利的地位。

1.2 本标准与国内外标准关系的说明

SPM 格式的技术标准化的趋势已经得到广泛认可，受到国内外各界的广泛关注。国际标准化组织(ISO)已经于 2004 年将 SPM 标准化列入其工作框架之内并建立了相应的技术委员会和分委员会(TC201/SC3)；建立了相应的研究小组(WG-1, Information formats)以深入研究 SPM 中的数据格式的标准化。按照 ISO 的规划，SPM 数据格式的标准化将以 SPM 术语为起点，按照数据传送、数据管理和数据存储的三步走策略逐步实现。SPM 术语是作为现有《表面化学分析标准》的补充内容发布的(ISO18115)。2011 年，ISO 发布了一个基于文本的、SPM 数据传送的格式标准，即 ISO 28600 (Data transfer Format)。根据 ISO 的技术路线，还将建立 SPM 的数据处理方法 (Data processing methods) 和 SPM 的综合数据库 (Comprehensive Database for SPM) 的技术标准。从国际国内的发展现状和趋势来看，建立一种适合用于对 SPM 仪器所产生的数据进行保存和分析处理的 SPM 数据存储格式方案，是非常迫切需要的。

2 标准起草工作简况

标准起草工作简况，包括任务来源、协作单位、主要工作过程、主要起草人及其所做的工作等。

2.1 任务来源

本任务来源于广东省分析测试协会团体标准制修订计划项目《表面化学分析 扫描探针显微镜数据存储格式》（2022年5月至2024年4月）。主导单位为中山大学，包括中山大学物理学院和中山大学分析测试中心。协作单位包括广州中源仪器技术有限公司、暨南大学理工学院。

2.2 起草单位、起草人

主要起草人及其所做的工作如下：

丁喜冬（中山大学物理学院）：负责数据存储格式的总体设计、初稿撰写、实现、测试与应用。

肖章武（广州中源仪器技术有限公司）：负责数据存储格式的范例实现、程序设计和测试、初稿的修订和应用。

赵亮兵（广州中源仪器技术有限公司）：负责数据存储格式的程序测试和标准的应用。

陈建（中山大学分析测试中心）：负责数据存储格式的的测试、修订和标准的应用。

龚力（中山大学分析测试中心）：负责标准的测试、比较分析和应用。

谢伟广（暨南大学理工学院）：负责标准范例及仪器产品的测试和应用。

刘金超（华南理工大学分析测试中心）：负责标准的测试、比较分析和应用。

2.3 主要工作过程

2022年5月：完成标准初稿的撰写；

2022年10月：完成标准的修改与提交，项目通过论证并正式立项；

2023年1月：启动标准的应用，用C语言编写程序，在广州中源仪器有限公司生产的扫描探针显微镜产品首次使用符合本标准的扫描探针显微镜数据存储格式，实现了扫描探针显微镜数据的存储，包括读、写和现实等功能。

2023年5月：完成标准的应用范例的编程，按照该标准编写了程序软件并应用于广州中源仪器有限公司生产的扫描探针显微镜产品中，并对程序进行了测试和试用。

2023年8月：根据仪器应用情况，完成了对标准的修订，形成了标准的讨论稿，在广州中源仪器有限公司、中山大学物理学院、中山大学分析测试中心进行了应用。

2023年10月：本标准的5家起草单位对标准讨论稿进行了讨论，广州中源仪器有限公司完成了产品范例的修改，改进后的标准范例和产品已在上述5家单位分别进行了进一步的测试和应用。

另外，在2022年5月之前，与本标准有关的相关工作如下：

本标准主要起草人项目团队核心成员（丁喜冬、陈建等）研究SPM实验数据的格式，主持制定了扫描探针显微镜的国家标准1件：中华人民共和国国家标准《表面分析化学：扫描探针显微镜数据传输格式》，GB/T 36052-2018，正式发布日期：2018.3.15。中山大学团队在国内最先对SPM数据格式进行了研究，分析了SPM数据格式标准化的要求和思路，提出了一种基于HDF5（HDF，Hierarchical Data Format）的SPM格式方案并开展了示例研究，发表了相关研究论文《探针显微镜中数据格式的分析及转换程序的设计》，电子显微学报，2011(03): 215-221。本项目研究团队致力于SPM仪器的高端产品的研发与应用。技术成果转化的主要承接单位是广州中源仪器技术有限公司。

3 标准编制原则和确定标准主要内容的论据

标准编制原则和确定标准主要内容的论据，标准修订项目还应当列出和原标准主要差异情况。

3.1 标准的编制原则

本标准依据《广东省分析测试协会团体标准制修订工作程序》、遵循“统一性、协调性、适用性、一致性、规范性”的原则，注重标准的可操作性，本标准按照 GB/T 1.1-2020《标准化工作导则 第 1 部分：标准的结构和编写规则》给出的规则进行编写和表述。

3.2 确定标准主要内容的依据

在表面分析测试领域，许多商用的扫描探针显微镜（SPM）在各种不同的环境下使用。SPM 的控制、数据的采集和处理都可以由带有存储器的计算机来实现，所有的数据及参数都记录在数字化的文件中。由于所用硬件平台不同，各制造商所生产的 SPM 大都使用专用的数据格式。这些数据格式大多互不兼容，给数据的存储、处理、传输、交换、共享等带来了不便。基于文本的 SPM 数据格式，虽然能够解决数据的传输、交换、共享等方面的问题，但存储效率不高且不利于数据的快速处理。各制造商所使用的 SPM 数据格式通常采用二进制存储，更加适合于数据的存储和分析处理，但存在兼容性、扩展性等方面的问题，不便于数据的传输、交换和共享。因此，需要一种用于 SPM 数据的存储和共享的标准格式，主要应用于 SPM 中的实时数据存储、高效分析处理和数据共享等计算处理领域。

建立 SPM 的数据存储格式的标准受到各界的重视和关注。SPM 数据传送的格式标准的出发点主要是实现不同制造商的仪器得到的数据的传输、交换、共享和归档，保证数据的完整性；增进交流，对从不同仪器得到的数据进行一致的解释和处理，以降低数据分析的不确定性。但 ISO 28600 并不是一套适合于对 SPM 数据进行高效存储和处理的格式方案。这是因为：该标准是中所有数据

都基于 ASCII 文本，与包含相同信息量的基于二进制的数据格式相比，占用的存储空间较大，因此不适合于海量数据的存储；其数据结构不支持随机读写，检索不便，进行数据的分析处理时效率不高；另外，由于格式固定，其扩充性受到一定制约。

本文件规定了在计算机上存储扫描探针显微镜（SPM）数据的一种标准格式。该格式以二进制和文本相混合的形式进行编码，将 SPM 采集的数据、参数和关联信息进行紧凑地存储，适合于对海量数据的存储和共享。该格式的数据结构支持随机读写，适合于对 SPM 数据进行高效的分析处理和读写。

4 方法验证及格式试验

方法验证及格式试验包括主要试验（或验证）的情况介绍、分析报告、相关技术和应用说明。

4.1 基本情况

为确保本标准方法格式的“统一性、协调性、适用性、一致性、规范性”，验证本标准数据存储格式的可操作性，标准起草工作小组组织 5 个单位（中山大学物理学院、广州中源仪器技术有限公司、中山大学分析测试中心、暨南大学理工学院、华南理工大学分析测试中心），利用多个扫描探针显微镜仪器平台对本标准中扫描探针显微镜数据存储格式进行了多方面的试验、验证以及测试和应用。

4.1.1 方法格式验证的内容

2023 年 5 月：广州中源仪器技术有限公司完成了标准应用范例的编程，在所生产的扫描探针显微镜仪器产品中使用了符合本标准所定义的数据存储格式。截止到 2023 年 10 月底，本标准的 5 家起草单位对标准进一步的测试和应用，并对本标准进行了讨论和改进。

5 家起草单位对本标准方法验证及格式试验的工作分别如下：

中山大学物理学院：负责数据存储格式的总体设计、初稿撰写、实现指导、在扫描探针显微镜仪器系统中本标准格式的试验与应用。

广州中源仪器技术有限公司：负责数据存储格式标准的范例实现、C++程序的设计和测试及在该公司仪器产品（扫描探针显微镜）中的功能实现、标准的应用结果收集及修改。

中山大学分析测试中心：负责仪器系统（扫描探针显微镜）中标准格式的分析测试、改进和应用。

暨南大学理工学院：负责仪器系统（扫描探针显微镜）中标准格式的试验和应用。

华南理工大学分析测试中心：负责仪器系统（扫描探针显微镜）中标准格式的分析测试和应用。

4.1.2 方法格式验证的结果

截止到 2023 年 10 月底，本团体标准《扫描探针显微镜数据存储格式》已在广州中源仪器技术有限公司制造的扫描探针显微镜产品中实现和应用，目前已经在广州中源仪器技术有限公司、中山大学物理学院、中山大学分析测试中心、暨南大学理工学院、华南理工大学分析测试中心多个单位完成验证和应用。结果表明，使用本数据存储格式标准的扫描探针显微镜产品的数据存储效率较高、性能稳定，仪器产品使用方便。建议进一步征求意见并发布和推广应用。

4.2 验证情况-1：广州中源仪器技术有限公司

4.2.1 基本情况

广州中源仪器技术有限公司主要完成本标准中格式中的各种数据类型的实现和验证，负责将 24 位编码格式和 32 位编码的数据存储格式标准分别集成到公司制造各型扫描探针显微镜仪器产品中，包括：负责数据存储格式标准的范例实现、C++程序的设计和测试及功能实现、标准的应用结果收集及修改。验证时，特别注重编程实现的方便性、简易性，并提供了相关的流程图、源代码和界面截图，对代码的运行情况进行了详细的分析和试验。

总之，本标准格式在广州中源仪器技术有限公司的扫描探针显微镜产品中已全面应用、测试和应用。结果表明，该格式数据存储效率较高、性能稳定。

方法验证及格式试验时所用系统的软硬件开发环境如下：

扫描探针显微镜的产品型号：中源仪器所有扫描探针显微镜产品，包括 WinSPM A/B/L、中源仪器 EDU 20。

验证格式类型：24 位编码格式标准、32 位编码格式标准

计算机硬件环境：品牌兼容机，Intel Core i5，16G 内存，SSD 256GB + HDD 1TB.

计算机软件环境：Windows 10，Visual Studio 2019 C#

4.2.2 验证过程及结果

1, 开发步骤

根据团体标准文件描述创建文件头部、各参数表表头结构定义源文件。

根据团体标准文件描述创建参数表结构描述 proto 文件。

使用 proto 工具根据描述文件生成 C#源文件实现参数表结构的序列化和反序列化。

在 SPM 在线控制软件工程中引入新生成的源文件，在相应的程序节点填入标准文件规定的信息，用户选择保存时将系统采集到的数字图像信息和谱图数据序列化后保存到磁盘文件。

2, 多通道图像数据保存流程:

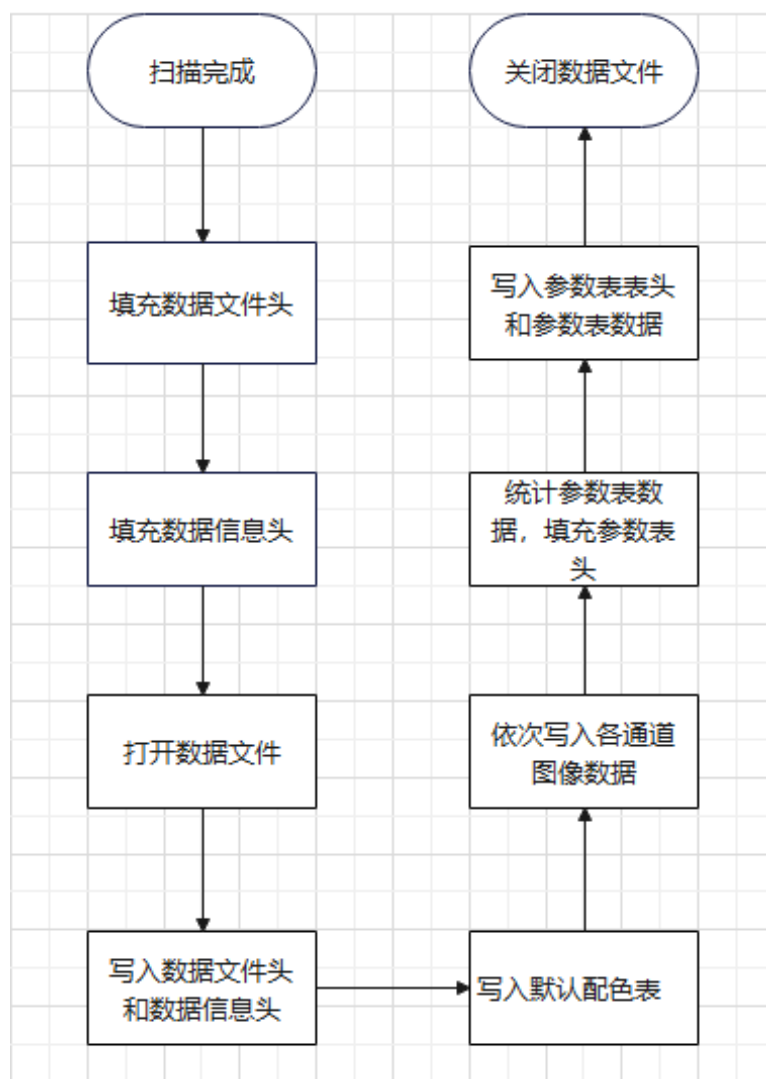


图 4.2-1 多通道图像数据保存流程

3, 谱测量数据保存流程:

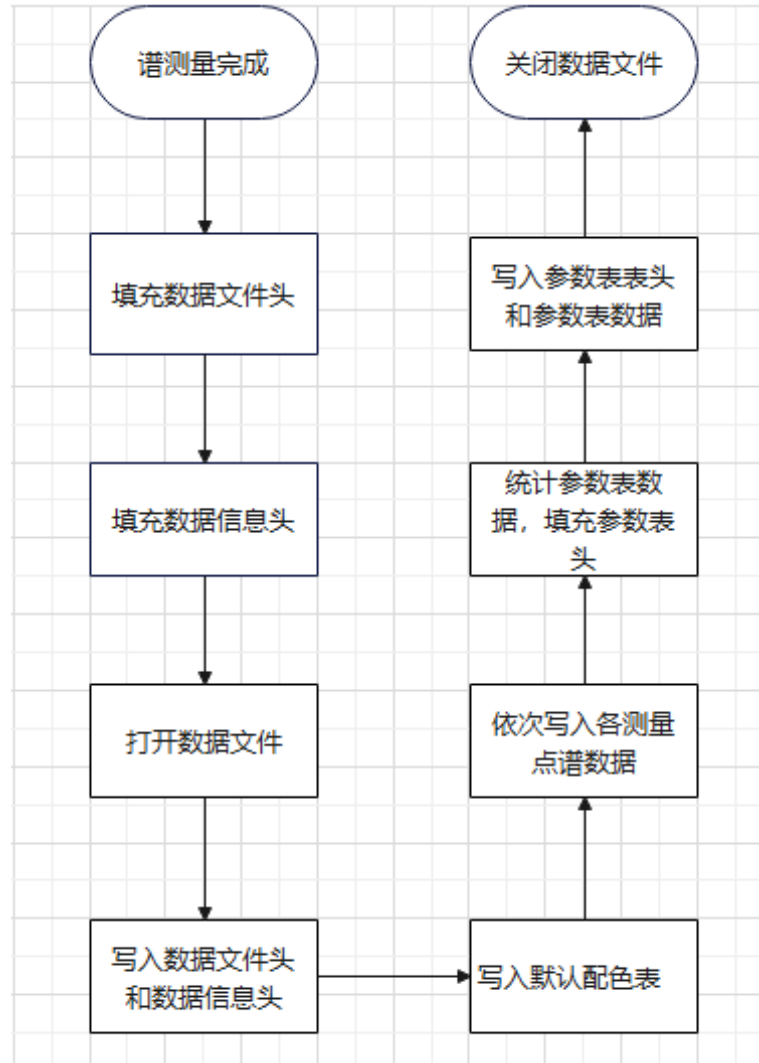


图 4.2-2 谱测量数据保存流程图

4. 文件头部定义源文件

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Runtime.InteropServices;
using System.Text;
using System.Threading.Tasks;

namespace SPMController
{
    [StructLayout(LayoutKind.Sequential, Pack = 1)]
    public struct SpmDataFileHeader
    {
        public UInt16 sfType;
```

```

    public Int32 sfSize;
    public Int32 sfDataType;
    public Int32 sfDataOffset;
}

[StructLayout(LayoutKind.Sequential, Pack = 1)]
public struct SpmDataInfoHeader
{
    public Int32 sfHeadSize;
    public Int32 sfWidth;
    public Int32 sfHeight;
    public UInt16 sfPlanes;
    public UInt16 sfBitCount;
    public Int32 sfCompression;
    public Int32 sfDataSize;
    public Int32 sfXScale;
    public Int32 sfYScale;
    public Int32 sfDataClrUsed;
    public Int32 sfDataImportant;
}

[StructLayout(LayoutKind.Sequential, Pack = 1)]
public struct SpmTotalParametersHeader
{
    public UInt32 saParametersIdentifier;           // 参数表的类型
    public Int32 saParametersSize;                 // 参数表字节数
    public Int32 saParametersNumber;              // 参数表项数量
    public UInt32 saMaxDataValue;                 // 位图数据的最大值
    public Int32 saBaseParametersOffset;          // 基本参数表的偏移量(从参数表
头开始)
    public Int32 saExtendParametersOffset;        // 扩展参数表的偏移量(从参数表
头开始)
    public Int32 saRelationParametersOffset;      // 关联参数表的偏移量(从参数
表头开始)
}

[StructLayout(LayoutKind.Sequential, Pack = 1)]
public struct SpmSubParametersHeader
{
    public UInt32 saParametersIdentifier;           // 参数表的类型
    public Int32 saParametersSize;                 // 参数表字节数
    public Int32 saParametersNumber;              // 参数表项数量
}

[StructLayout(LayoutKind.Sequential, Pack = 1)]
public struct SpmSpectraHeader
{
    public UInt32 saParametersIdentifier;           // 参数表的类型
    public Int32 saParametersSize;                 // 参数表字节数
    public Int32 saSpectraNumber;                 // 谱曲线显示信息表数据项数量
    public Int32 saOrdinatesNumber;               // 纵坐标(待测变量)表项数量
}

```

```

    public Int32 saTimesNumber;          // 每个位置测量遍次数
    public Int32 saPositionsNumber;      // 测量位置表项数量
    public Int32 saDisplayColorUsed;     // 谱曲线数据在显示时使用的颜色
数目
    public Int32 saDisplayColorEach;     // 每条谱曲线数据在显示时使用的
颜色数目
    public Int32 saPaletteIdentifier;    // 对应彩色表索引号
    public Int32 saPaletteNumber;       // 对应彩色表颜色数量
}

[StructLayout(LayoutKind.Sequential, Pack = 1)]
public struct SpmDataRelationHeader
{
    public UInt32 saParametersIdentifier; // 参数表的类型
    public Int32 saParametersSize;       // 插件表字节数
    public Int32 saParametersNumber;     // 插件表数据的项数
    public Int32 saParametersOffset;     // 插件表数据的偏移量(从关联表头开始)
    [MarshalAs(UnmanagedType.ByValArray, SizeConst = (16))]
    public byte[] saDataIdentifier;      // 数据的唯一标识
    [MarshalAs(UnmanagedType.ByValArray, SizeConst = (16))]
    public byte[] saAuxiliaryIdentifier;  // 数据的辅助标识
    public UInt16 saSubdataCoordinates1;
    public UInt16 saSubdataCoordinates2;
    public UInt16 saSubdataCoordinates3;
    public UInt16 saSubdataCoordinates4;
}
}

```

5. 各参数表表结构定义源文件（部分）

```

// <auto-generated>
//     Generated by the protocol buffer compiler.  DO NOT EDIT!
//     source: proto/SPMDataParameters.proto
// </auto-generated>
#pragma warning disable 1591, 0612, 3021, 8981
#region Designer generated code
using pb = global::Google.Protobuf;
using pbc = global::Google.Protobuf.Collections;
using pbr = global::Google.Protobuf.Reflection;
using scg = global::System.Collections.Generic;
namespace SPMController.ProtoData {
    /// <summary>Holder for reflection information generated from
    proto/SPMDataParameters.proto</summary>
    public static partial class SPMDataParametersReflection {
        #region Descriptor
        /// <summary>File descriptor for proto/SPMDataParameters.proto</summary>
        public static pbr::FileDescriptor Descriptor {
            get { return descriptor; }
        }
        private static pbr::FileDescriptor descriptor;
        static SPMDataParametersReflection() {

```

```

byte[] descriptorData = global::System.Convert.FromBase64String(
    string.Concat(
        "Ch1wcm90by9TUE1EYXRhUGFyYW1ldGVycy5wcm90bXlXU1BNQ29udHJvbGx1",
        "ci5Qcm90b0RhdGEiJAoRU3BtQmFzZVBhcmFtZXR1cnMSDwoHc3BMYWJ1bGgB",
        "IAMoCSKHAQoWU3BtRXhwZXJpbWVudFBhcmFtZXR1chIPCgdzcExhYmVsGAEg",
        "ASgJEg4KBnNwVW5pdBgCIAEoCRIPCgdzcFZhbHVlGAMgASgBEhUKDXNwQ2Fs",
        "aWJyYXRpb24YBCABKAESFgoJc3BDb21tZW50GAUgASgJSACIAQFCDAoKX3Nw",
        "Q29tbWVudCJYChdTcG1FeHB1cm1tZW50UGFyYW1ldGVycxI9CgRkYXRhGAEg",
        "AygLmi8uU1BNQ29udHJvbGx1ci5Qcm90b0RhdGEuU3BtRXhwZXJpbWVudFBh",
        "cmFtZXR1ciLiAgoPU3BtSW1hZ2VEaXNwbGF5Eg8KB3NwTGFiZWwYASABKAKS",
    )
);

```

6. 参数表结构描述 proto 文件（部分）

```

syntax = "proto3";
option cc_enable_arenas = true;

package SPMController.ProtoData;

message SpmBaseParameters {
    repeated string spLabel = 1;
}

message SpmExperimentParameter {
    string spLabel = 1;
    string spUnit = 2;
    double spValue = 3;
    double spCalibration = 4;
    optional string spComment = 5;
}

message SpmExperimentParameters {
    repeated SpmExperimentParameter data = 1;
}

message SpmImageDisplay {
    string spLabel = 1;
    string spUnit = 2;
    int32 spChannelNo = 3;
    int32 spPassNo = 4;
    double spDataStart = 5;
    double spDataEnd = 6;
    double spDisplayStart = 7;
    double spDisplayEnd = 8;
    double spImportantStart = 9;
    double spImportantEnd = 10;
    int32 spDisplayColorUsed = 11;
    int32 spDisplayColorImportant = 12;
    int32 spPaletteIndex = 13;
    int32 spPaletteNumber = 14;
    string spComment = 15;
}

```


7. 测试验证步骤:

运行 SPM 在线控制软件。

连接到 WinSPM 原子力显微镜仪器，在“设置”-“常用设置”中选择保存为团体标准格式。

将合适的样品放置在样品台，执行调节、逼近等准备工作。

打开扫描模块，设置需要的扫描范围、扫描分辨率、成像通道，选择自动保存，点击“运行”开始扫描样品。

程序在扫描完成后自动保存。

8. 测试验证结果:

表 4.2-1 数据文件头验证情况

字段序号	字段标识	字段名称	验证情况
1	bitmap file identifier	数据文件类型标识符	通过
2	file Size	数据部分的大小	通过
3	data type identifier	SPM 图像数据的类型标识符	通过
4	Bitmap Data Offset	图像数据偏移量	通过

表 4.2-2 数据信息头验证情况

字段序号	字段标识	字段名称	验证情况
1	bitmap head Size	数据信息头的长度	通过
2	bitmap width	位图宽度	通过
3	bitmap height	位图高度	通过
4	bitmap planes	位图的位面数	通过
5	bit count	像素数据位数	通过
6	data compression	数据的压缩方式	通过
7	bitmap data Size	位图数据的大小	通过
8	bitmap X scale	位图的水平分辨率	通过
9	bitmap Y scale	位图的垂直分辨率	通过
10	color used	使用的颜色索引数	通过
11	color Important	重要的颜色索引数	通过

表 4.2-3 参数表表头验证情况

字段序号	字段标识	字段名称	验证情况
1	Total Parameters Identifier	参数表标识符='PARS'	通过
2	Total Parameters Size	整个参数表的字节数	通过
3	Total Parameters Number	整个参数表数据的个数	通过
4	Bitmap Data Maximum Value	位图数据最大值	通过
5	Base Parameters Offset	从参数表头开始到基本参数表之间的字节的偏移量	通过
6	Extend Parameters Offset	从参数表头开始到扩展参数表开始之间的字节的偏移量	通过
7	Relation Parameters Offset	从参数表头开始到关联参数表开始之间的字节的偏移量	通过

表 4.2-4 基本参数表表头验证情况

字段序号	字段标识	字段名称	验证情况
1	Base Parameters Identifier	参数表标识符='BASE'	通过
2	Base Parameters Size	基本参数表的字节数	通过
3	Base Parameters Number	基本参数表数据的个数	通过

表 4.2-5 扩展参数表表头验证情况

字段序号	字段标识	字段名称	验证情况
1	Extend Parameters Identifier	扩展参数表标识符='EXTD'	通过
2	Extend Parameters Size	扩展参数表的字节数	通过
3	Extend Parameters Number	扩展参数表数据的个数	通过

表 4.2-6 关联参数表表头验证情况

字段序号	字段标识	字段名称	验证情况
1	Relation Parameters Identifier	关联参数表标识符='RELA'	通过
2	Relation Parameters Size	关联参数表的字节数	通过
3	Relation Parameters Number	关联参数表数据的个数	通过
4	Data Identifier	SPM 数据的唯一标识	通过

5	Auxiliary Identifier	SPM 数据的辅助标识	通过
6	Subdata Coordinates 1	子数据坐标 1 的最大数	通过
7	Subdata Coordinates 2	子数据坐标 2 的最大数	通过
8	Subdata Coordinates 3	子数据坐标 3 的最大数	通过
9	Subdata Coordinates 4	子数据坐标 4 的最大数	通过

表 4.2-7 数据来源表表头验证情况

字段序号	字段标识	字段名称	验证情况
1	Parameters List Identifier	数据来源表标识符='DTSR'	通过
2	Relation Parameters Size	数据来源表的字节数	通过
3	Relation Parameters Number	数据来源表数据的个数	通过

表 4.2-8 插件信息表表头验证情况

字段序号	字段标识	字段名称	验证情况
1	Parameters List Identifier	插件信息表标识符='PLUG'	通过
2	Plugin Parameters Size	插件信息表的字节数	通过
3	Plugin Parameters Number	插件信息表数据的个数	通过

表 4.2-9 关联参数表表头验证情况

字段序号	字段标识	字段名称	验证情况
1	Parameters List Identifier	处理参数表标识符='TRMT'	通过
2	Relation Parameters Size	处理参数表的字节数	通过
3	Relation Parameters Number	处理参数表数据的个数	通过

下面给出了一些与测试验证结果相关的软件测试过程的截图。

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00000000	42	4D	36	02	08	00	4D	50	4D	43	36	02	00	00	28	00
00000010	00	00	00	01	00	00	00	FF	FF	01	00	20	00	00	00	00
00000020	00	00	00	00	00	00	8E	37	00	00	8E	37	00	00	80	00
00000030	00	00	80	00	00	00	00	00	00	00	00	00	00	00	00	00
00000040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000050	2B	00	00	00	00	00	00	00	31	00	00	00	00	00	00	00
00000060	37	0B	00	00	3A	0E	00	00	3D	11	00	00	40	14	00	00

图 4.2-3 多通道图像数据文件头文件截图

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	42	4D	36	02	08	00	4D	50	4D	43	36	02	00	00	28	00	BM6 MPMC6 (
00000010	00	00	00	01	00	00	00	FE	FF	FF	01	00	20	00	00	00	ÿÿ
00000020	00	00	00	00	08	00	8E	37	00	00	8E	37	00	00	80	00	!7 !7 !
00000030	00	00	80	00	00	00	00	00	00	00	00	00	00	00	00	00	!
00000040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000050	20	00	00	00	2E	00	31	00	00	00	34	08	00	00	00	00	+ . 1 4
00000060	37	0B	00	00	3A	0E	00	00	3D	11	00	00	40	14	00	00	7 : = @
00000070	43	17	00	00	46	1A	00	00	49	1D	00	00	4C	20	00	00	C F I L
00000080	4F	23	00	00	52	26	00	00	55	29	00	00	58	2C	00	00	O# R& U) X,
00000090	5B	2F	00	00	5E	32	00	00	61	35	00	00	64	38	00	00	[/ ^2 a5 d8

图 4.2-4 多通道图像数据信息头文件截图

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00080230	00	00	9D	83	00	00	50	41	52	53	24	04	00	00	81	00	! PARSS
00080240	00	00	FF	FF	00	00	1C	00	00	00	E2	02	00	00	E4	03	ÿÿ ä ä
00080250	00	00	42	41	53	45	BA	02	00	00	7F	00	00	00	0A	23	BASE#
00080260	49	53	4F	2F	54	43	20	32	30	31	20	53	50	4D	20	44	ISO/TC 201 SPM D
00080270	61	74	0A	13	67	65	6E	65	72	61	6C	20	69	6E	66	66	Transfer For
00080280	6D	61	74	0A	13	67	65	6E	65	72	61	6C	20	69	6E	66	mat general inf
00080290	6F	72	6D	61	74	69	6F	6E	0A	24	E5	B9	BF	E5	B7	9E	ormation \$â'¿â·!
000802A0	E4	B8	AD	E6	BA	90	E4	BB	AA	E5	99	A8	E6	8A	80	E6	ä, -æº ä»ââ!æ æ
000802B0	9C	AF	E6	9C	89	E9	99	90	E5	85	AC	E5	8F	B8	0A	0F	!æ é! â!-â ,
000802C0	E6	BF	80	E5	85	89	E6	A3	80	E6	B5	8B	41	46	4D	0A	æ¿!â æf æµ AFM

图 4.2-5 多通道图像参数表头文件截图

4.3 验证情况-2: 中山大学物理学院

4.3.1 基本情况

中山大学物理学院负责数据存储格式的总体设计、初稿撰写、实现指导、在扫描探针显微镜仪器系统中本标准格式的试验与应用。在方法验证及格式试验方面，中山大学物理学院主要完成了本标准中格式中 24 位编码的格式类型的编程实现、测试、试验和应用，以及 32 位编码格式标准的测试验证和应用，内容包括测试环境、过程、方法、结果和结论。

总之，本标准格式在中山大学物理学院已进行全面的测试、试验和应用。结果表明，应用该格式的扫描探针显微镜系统性能稳定、使用方便。

方法验证及格式试验时所用仪器系统的软硬件开发环境如下：

计算机硬件环境：Dell 台式机，Intel Core i5，16G 内存，SSD 256GB + HDD 1TB.

扫描探针显微镜的产品型号：WinSPM EDU 20-A

探针类型：Akiyama-probe

验证格式类型：24 位编码的格式类型、32 位编码格式标准

4.3.2 验证过程及结果

在上述扫描探针显微镜产品上进行方法格式验证，采用 Akiyama 压电自感应探针，对各种样品进行扫描成像，本标准中格式中 24 位编码的格式类型进行数据的存储和读出，以及 32 位格式标准的使用。图像扫描的验证结果如下。

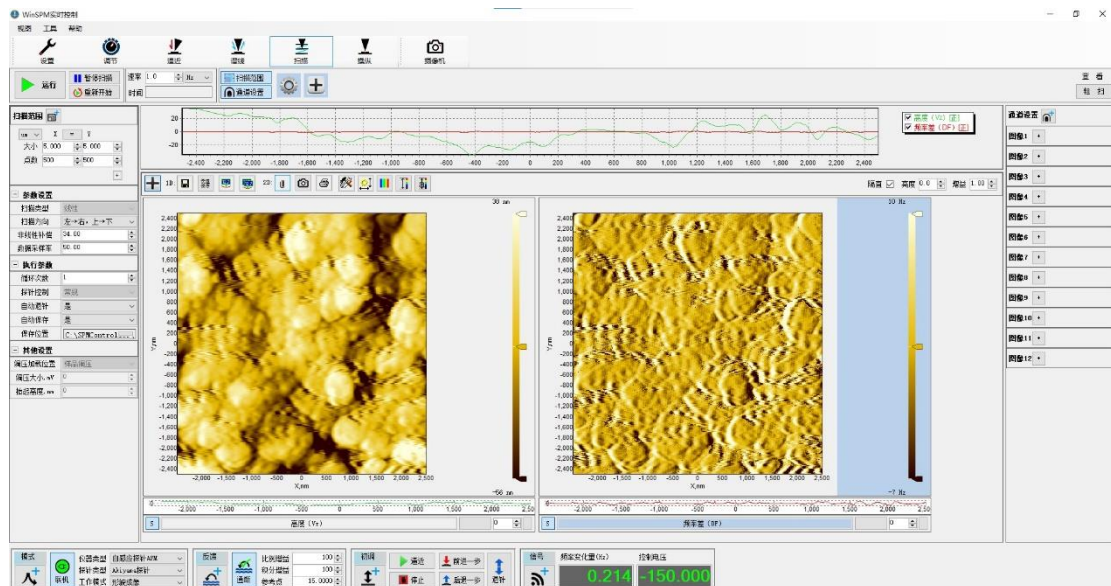


图 4.3-1 磁盘形貌扫描，采用 24 位编码的格式

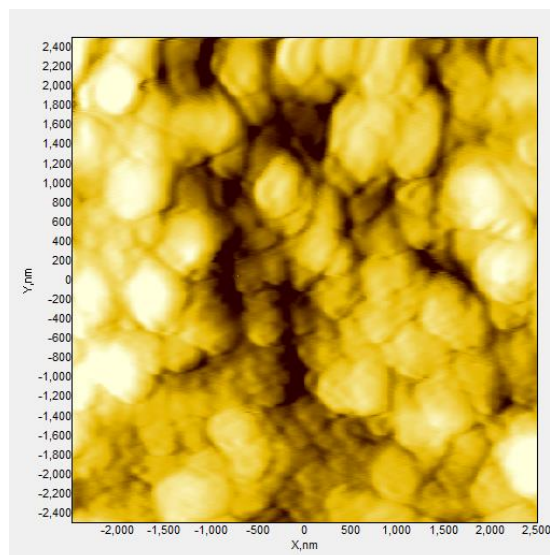


图 4.3-2 存储后读出的磁盘形貌图，采用 24 位编码的格式

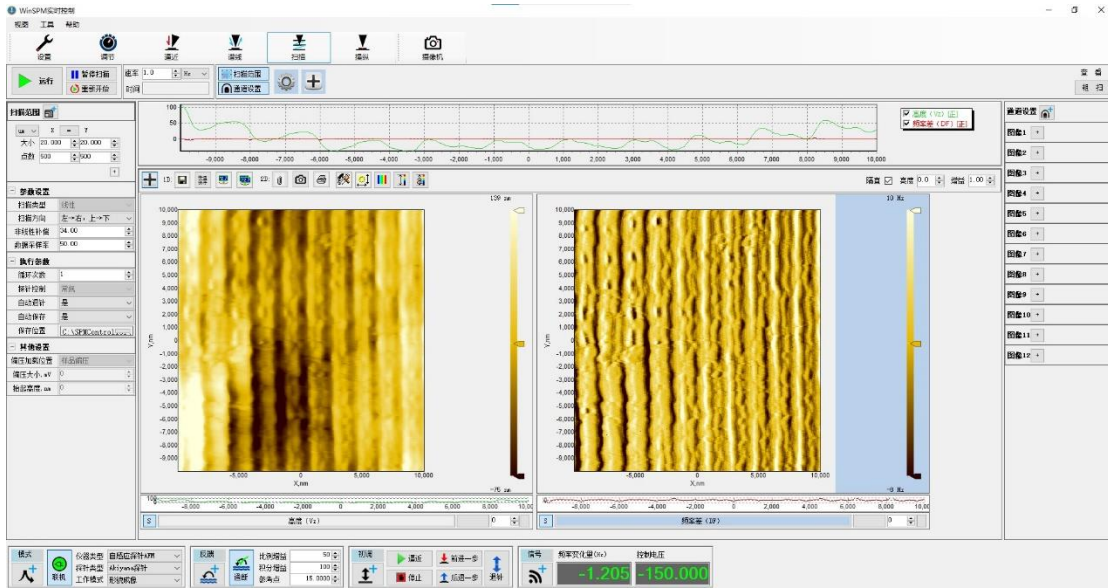


图 4.3-3 光栅形貌扫描，采用 32 位编码的格式类型

4.4 验证情况-3：中山大学分析测试中心

4.4.1 基本情况

中山大学分析测试中心负责仪器系统（扫描探针显微镜）中标准格式的分析测试、改进和应用。在方法验证及格式试验方面，中山大学分析测试中心主要完成了 24 位编码的格式类型、32 位编码格式标准的测试和验证（包括图像和谱数据），使用音叉型自感应探针和激光探针，funny 光栅、磁盘等样品。

总之，中山大学分析测试中心对本标准格式进行了分析测试、试验和应用。结果表明，本标准格式及其仪器系统使用方便、性能稳定。

方法验证所用仪器系统的软硬件环境如下：

计算机硬件环境：品牌兼容机，Intel Core i5，16G 内存，SSD 256GB + HDD 1TB.

扫描探针显微镜产品型号：WinSPM B

探针类型：Akiyama-probe、激光探针

验证格式类型：24 位编码的格式类型、32 位编码格式标准

4.4.2 验证过程及结果

在上述扫描探针显微镜产品上进行方法格式验证，采用 Akiyama 压电自感应探针，对各种样品进行扫描成像，本标准中格式中 24 位编码的格式类型进行数据的存储和读出，以及 32 位格式标准的使用。图像扫描的验证结果如下。

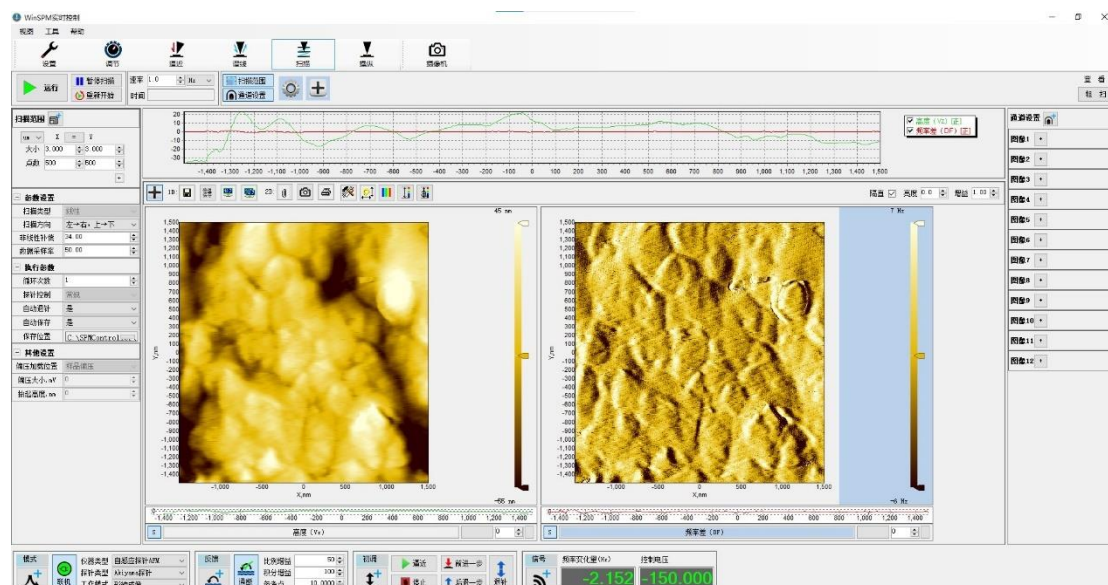


图 4.4-1 磁盘形貌扫描，采用 24 位编码的格式

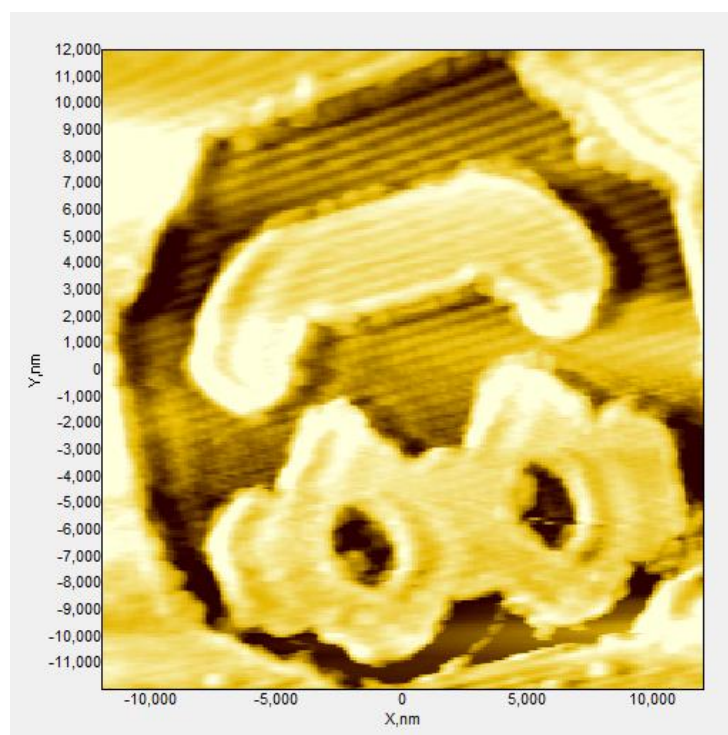


图 4.4-2 存储后读出的光栅 funny 图案，采用 32 位编码的格式标准

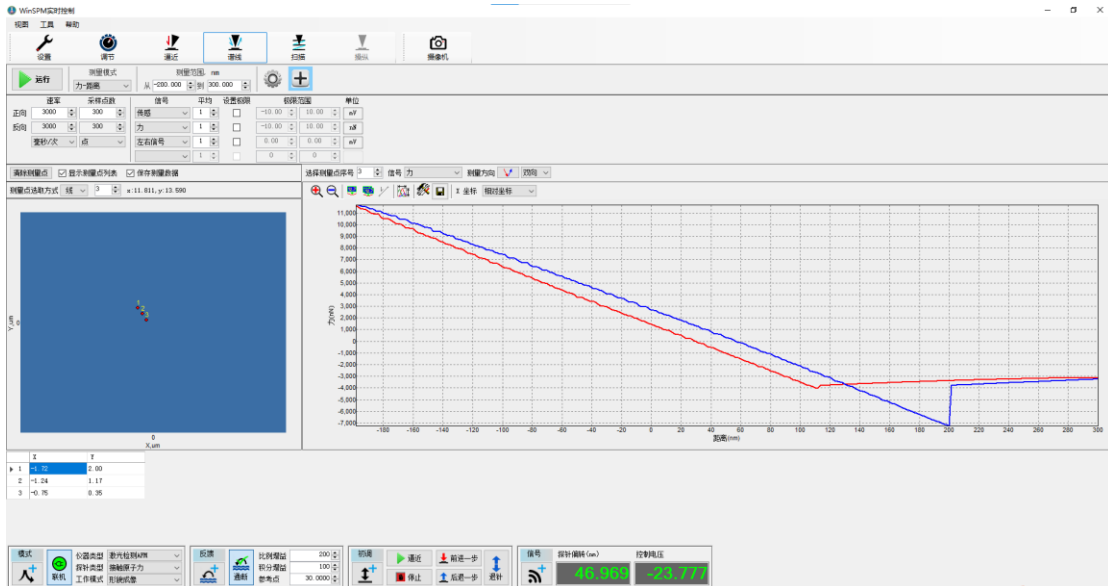


图 4.4-3 谱数据（力-距离曲线），采用 32 位编码的格式类型

4.5 验证情况-4：暨南大学理工学院

4.5.1 基本情况

在方法验证及格式试验方面，暨南大学理工学院主要负责仪器系统（扫描探针显微镜）中标准格式的试验和应用。暨南大学理工学院使用激光型探针和音叉型自感应探针完成了 32 位编码格式标准的图像数据的测试和验证，样品为磁盘、funny 图案和光栅。

总之，暨南大学理工学院对本标准格式进行了试验和应用。结果表明，所使用的扫描探针显微镜仪器性能稳定，本标准格式使用和应用方便。

方法验证所用仪器系统的软硬件环境如下：

计算机硬件环境：dell 兼容机，Intel Core i5，16G 内存

扫描探针显微镜产品型号：WinSPM B

探针类型：轻敲模式激光探针，Akiyama-probe

验证格式类型：32 位编码格式标准的图像数据

4.5.2 验证过程及结果

在上述扫描探针显微镜产品上进行方法格式验证，分别采用激光探针和 Akiyama 压电自感应探针，对各种样品进行扫描成像，用本标准中格式中的 32 位格式标准存储和读出图像数据。

存储和读出图像的验证结果如下。

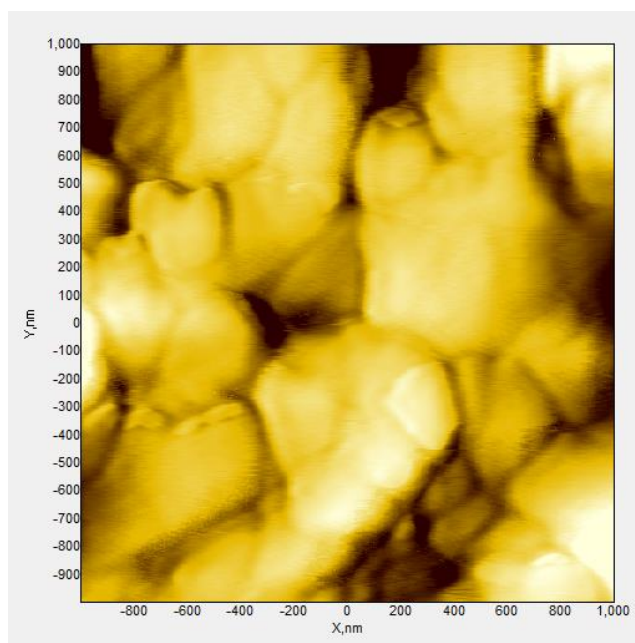


图 4.5-1 磁盘形貌图，轻敲模式激光探针，采用 32 位编码的格式标准

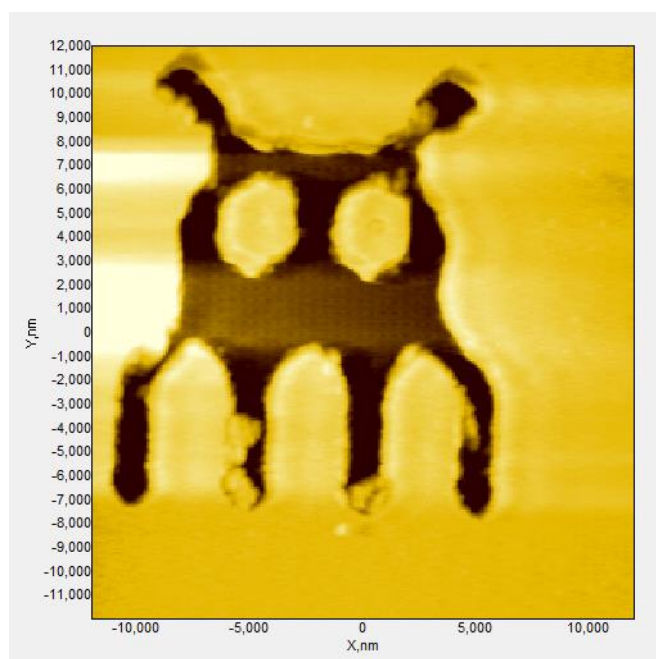


图 4.5-2 光栅 funny 图案，Akiyama-probe，采用 32 位编码的格式标准

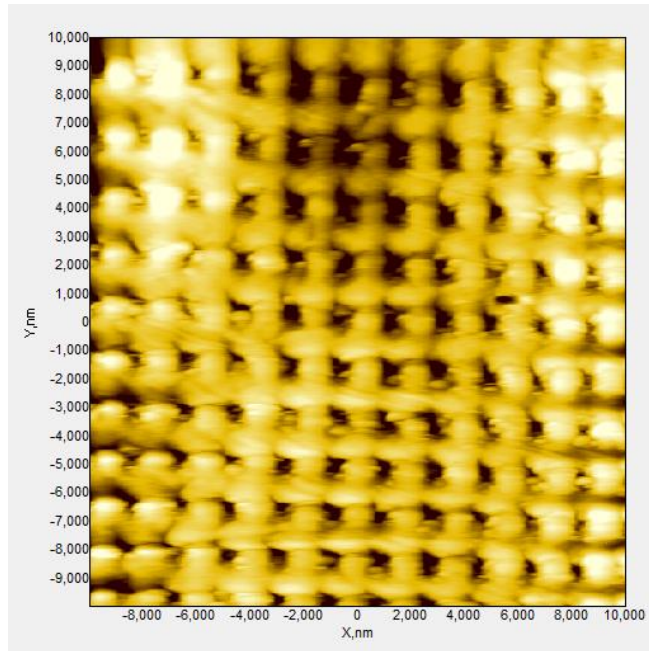


图 4.5-3 二维光栅图案，Akiyama-probe，采用 32 位编码的格式标准

4.6 验证情况-5：华南理工大学分析测试中心

4.6.1 基本情况

在方法验证及格式试验方面，华南理工大学分析测试中心主要负责仪器系统（扫描探针显微镜）标准格式的分析测试和应用。华南理工大学分析测试中心使用激光型探针的接触模式形貌成像、完成了 32 位编码格式标准的图像数据的测试和验证，样品为光盘。

总之，华南理工大学分析测试中心对本标准格式进行了测试和应用。结果表明，所使用的扫描探针显微镜系统应用了本标准格式，仪器使用方便。

方法验证所用仪器系统的软硬件环境如下：

计算机硬件环境：dell 兼容机，Intel Core i5，16G 内存

扫描探针显微镜产品型号：WinSPM L

探针类型：接触模式的激光型探针

验证格式类型：32 位编码格式标准的图像数据

4.6.2 验证过程及结果

在上述扫描探针显微镜产品上进行方法格式验证，采用激光探针，对各种样品进行扫描成像，用本标准中格式中的 32 位格式标准存储和读出图像数据。

扫描和存储的图像的验证结果如下。

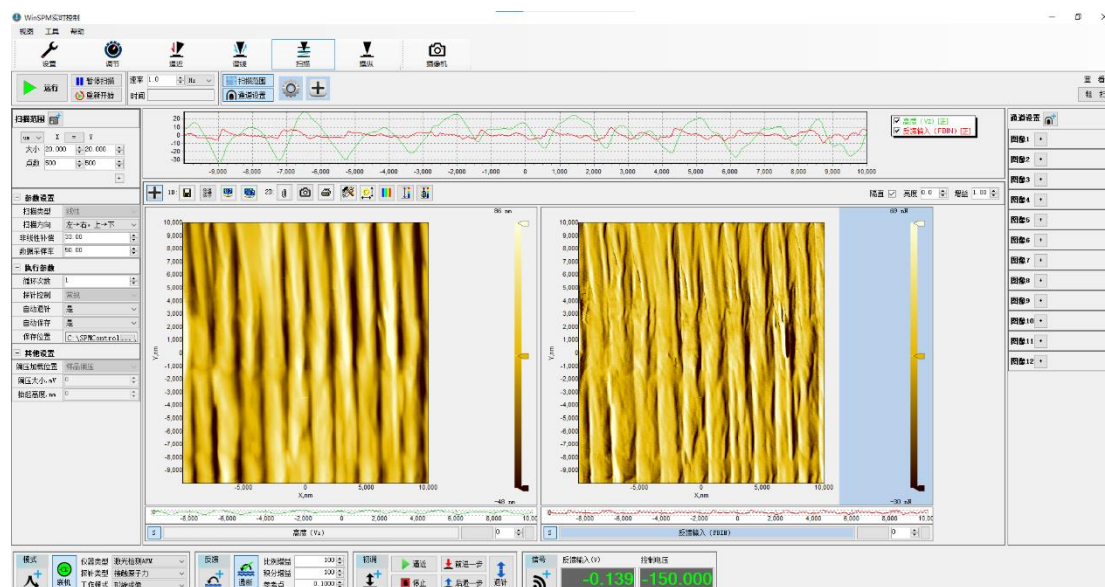


图 4.6-1 光盘形貌图，接触模式激光探针，采用 32 位编码的格式标准

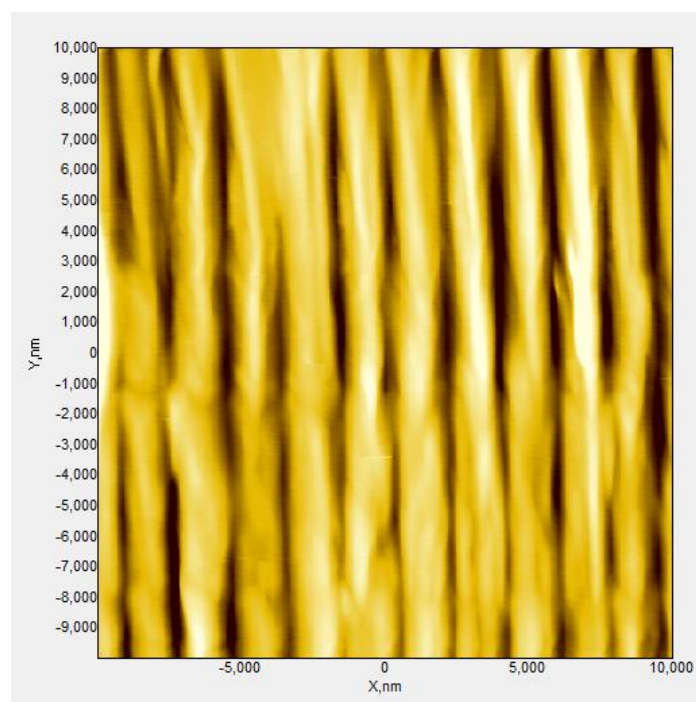


图 4.6-2 存储和读出后的光盘形貌图，激光探针，采用 32 位编码的格式标准

5 采用国际标准和国外先进标准情况及水平对比

本标准的技术关键在于精准把握国内扫描探针显微镜（SPM）仪器制造商的产品特点和应用领域使用者的实际需求，以及国际 SPM 技术及其应用的发展趋势。本标准重点研究了如何兼顾数据的存储效率和处理效率的、SPM 的数据计算机存储的技术方案，分析 SPM 数据存储和分析处理的未来技术的发展思路，确定 SPM 数据存储格式标准的建立原则和优化思路，评估其效率、兼容性、以及其开放性、可扩展性，最终综合确定 SPM 数据存储的格式和标准，填补我国在该 SPM 标准领域的空白，推动与国际标准的接轨和实质性的参与国际竞争。

本标准的格式是为 SPM 的数据存储而设计的，包括扫描隧道显微镜（STM）、原子力显微镜（AFM）、静电力显微镜（EFM）、磁力显微镜（EFM）等，以及其它采用尖锐针尖在样品表面扫描的相关 SPM 表面分析方法。本格式涵盖了 SPM 的单通道成像、多通道成像、单点谱和多点谱的数据。

SPM 数据存储格式的基本构想是该格式用普通计算机可读、可写、可共享；为了紧凑存储数据、减小存储空间、利于海量存储，该格式中的数据主要采用基于二进制的存储方式；为了实现对数据的高效分析处理，该格式的数据结构应支持数据的随机读写。

为了有足够的灵活性以适应未来 SPM 衍生仪器的扩展、有足够的通用性以适用各种被测物理量，本格式中的数据参数部分兼容现有的表面化学分析的国际标准 ISO/DIS 28600:2011 Surface chemical analysis - Data transfer format for scanning probe microscopy，或国家标准 GB/T 36052-2018 表面化学分析 - 扫描探针显微镜数据传送格式。本标准主要解决了 SPM 数据格式的存储效率的问题。

6 与现行法律、法规、政策及相关标准协调性

该标准项目目前没有对应的国际标准或国外标准。

目前 SPM 内部数据的存储格式大多采用各厂商自行定义的专用的数据格式，并没与统一的格式标准。现在行业内几十种 SPM 格式，下表 1 列示了国内外几种有代表性的格式。从表中可以看出，目前各标准在格式名，格式后缀，开放性及文件存储类型上有很大的不同。本项目 SPM 数据存储格式标准的确立，需要对各数据格式的结构进行深入研究，以便尽量满足本项目数据格式标准的兼容性和普适性。

表 1 现有主要 SPM 数据格式

格式名称	制定方	格式后缀	开放性与否	文件类型
ISO 标准	ISO	*.txt	是	文本
Nanoscope	Bruker 公司	*[0-9]{3}	否	混合
SM3	RHK 公司	*.sm3	否	二进制
GXSM	unidata	*.nc	是	二进制
SPML	Twente 大学	*.spml	是	文本
NSPM	中山大学	*.[a-z0-9]{3}	是	混合
CSPM	本原公司	*.csm	是	混合

7 公开标准涉及专利的信息

本标准是在本项目团队长期开展 SPM 技术及其应用研究的基础上，经过对 SPM 数据存储格式的研究并经应用，并与本行业企业合作的基础上联合提出的，将具有完整的自主知识产权。

8 重大意见分歧的处理依据和结果

本标准的编写过程中无重大意见分歧。

9 预期的社会效益及贯彻实施标准的要求、措施等建议

建议将本标准在国内外相关仪器制造厂商和科研机构进行推广应用，并进一步争取成为国家标准或行业标准，推动与国际标准的接轨和实质性参与国际竞争。

10 其他应当说明的事项

无。