



secsgem s1f3



云中飞鸿--\*\*峰

已关注

# 半导体设备通讯标准(SECS/GEM)

转载 云中飞鸿--\*\*峰 已于 2025-02-28 18:13:34 修改 阅读量605 收藏3 点赞数  
分类专栏: 半导体相关 文章标签: 网络

CC

半导体相关 专栏收录该内容

3 篇文章

原文链接: <https://zhuanlan.zhihu.com/p/591684532>

这篇文章写的不错, 转载记录一下, 仅供学习参考, 如有侵犯权益, 可联系删除!

## 前言

在高度自动化的半导体制造厂中, **CIM(Computer Integrated Manufacturing【计算机整合制造】)** 统一管理各设备的生产流程, 并随时监控设备过程以减少过程失误进而降低成本及提升产品的质量。但随着过程的不同, 各设备有着不同特性的差异且各制造商所提供的设备也不尽相同, 因此增加**CIM管理** 的困难与复杂程度。**软件集成自动化存在的主要问题是在不同的设备供应商之间没有标准的通讯协议。**设备供应商不向半导体生产商开放通讯软件, 这使得半导体生产商不得不建立他们自己的软件“连接”, 导致了项目费用的巨大增加。

**SEMI(Semiconductor Equipment and Materials Institute)【半导体设备材料国际联盟】** 制定了半导体设备通讯标准接口**SECS(Semiconductor Equipment Communication Standard【半导体设备通信标准】)**, 让CIM与设备间有通用的通讯标准接口, 设备制造商只要提供符合通讯标准规范的设备, 便可在CIM的管理系统, 不但可缩短设备开发的时间及成本, 并可增加设备装机的效率达到快速量产, 进而提升产能输出。

## 一、SECS/GEM 简介

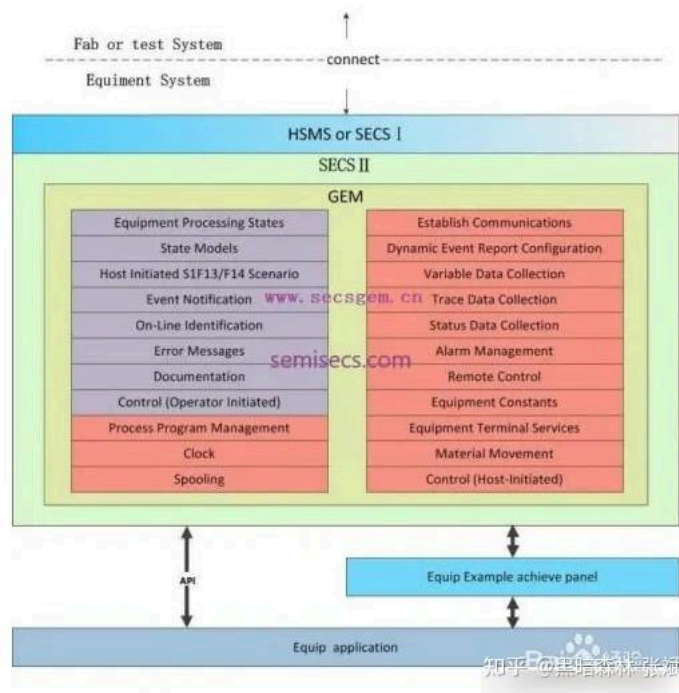
(Semiconductor Equipment Communication Standard/Generic Equipment Model)

SEMI (国际半导体产业协会) 是一个全球性的产业协会, 效劳于半导体、光伏 (PV)、电子等高科技行业的制造供给链。SEMI管理着SEMI国际规范创立了SEMI 自动化规范, 这是行业内供给商和客户之间不时开展的技术协议汇合。

SECS/GEM是由国际半导体设备与材料协会 (SEMI) 的会员一起构建的连接性标准。该标准应用于在设备和工厂管理系统间的通讯标准。

SECS/GEM是半导体的设备接口协议, 用于设备到主机的数据通讯。它通常用于半导体, TFT-LCD和电子行业, 由于这些行业的设备大多来自不同的乏统一的通讯规范。经过引入SECS/GEM, 制造可以搜集更多的信息, SECS/GEM使设备和主机中心可以畅通无阻地停止通讯, 从而完成智能工厂自

SECS/GEM协议网站 <http://www.secsgem.cn>



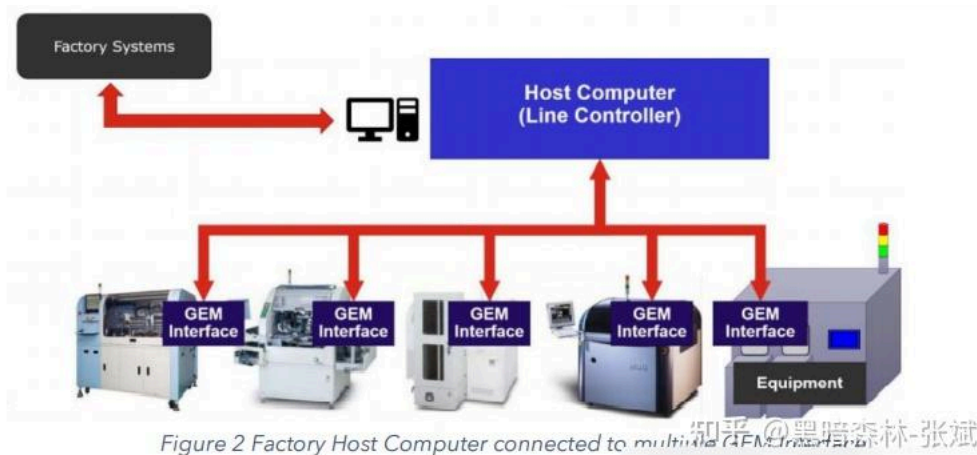


Figure 2 Factory Host Computer connected to multiple GEM Interfaces

SECS协议为点对点协议，它包括2个部分SECS- I 和SECS-II。

①SECS- I 为基于RS232的传输层，定义了设备和主机之间的MESSAGE交互的通信接口，大致相当于ISO/OSI模型的下面5层。主要包括有块传输协议MESSAGE接收算法和节点传输算法；

②SECS-II则把SECS- I 传输的二进制串翻译成形象直观的格式表示出来，SECS-II规范传输资料的标准结构和显示内容，方便使用者查看数据；

③SECS标准的典型应用，节点C代表主机，节点A和B表示设备，这些节点仅仅是组成整个网络的一部分。按照SECS标准的定义，每一个节点都能通过头中的设备号位决定向哪一个节点传递消息，根据回答位决定控制消息的传递方向是沿着树向下还是向上。节点A控制着1A、2A、3A 3个子节点，可的不同要求向节点C发送请求或者从节点C接收指令。SECSLine则表示了在2个节点之间按照SECS标准的电气接口。

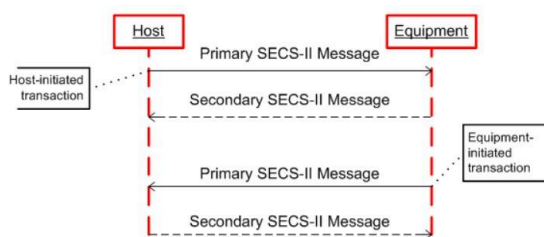


Figure 3 SECS-II Message Transactions

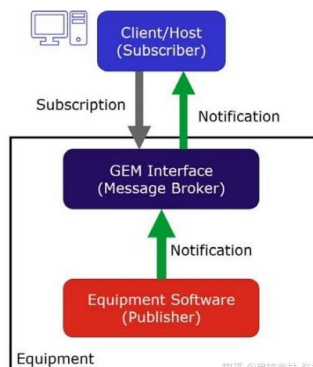


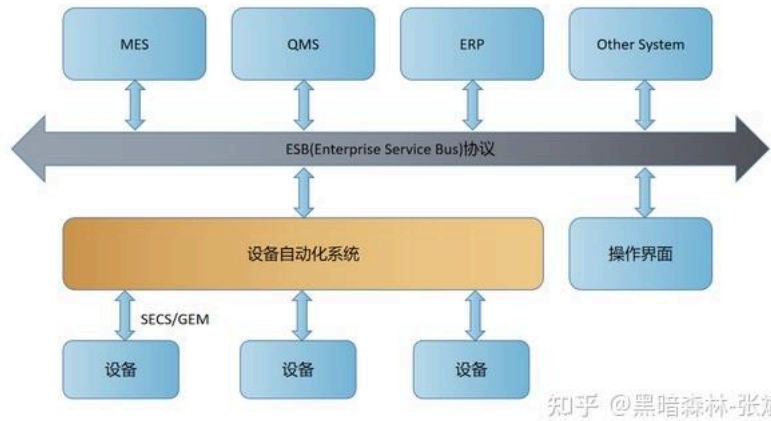
Figure 4 Message Broker

## 二、系统的硬件组成

整个控制系统由生产线上的生产设备和车间的控制系统组成，其硬件组成如图2所示。包括3个部分，设备和单元控制器之间通过SECS- I 连接，其他网相连。

处于最下方的是生产流水线，单元控制器与线上每道工序相应的控制软件负责控制流水线的生产，并将生产数据通过单元控制器上报给车间的MES系间管理人员的生产指令。

中间是MES系统，主要负责监控和控制流水线的生产情况、存储重要的生产数据、报表等，同时给操作管理人员提供一套管理系统来协调整个车间的最上层是工厂生产计划管理系统，用来给更高层的管理人员提供管理上的方便。



### 三、

#### 1、secs是一个协议族

#### 2、secs的消息通信

secs的通信玩法: secs通信的设备端通常为客户端 (Equipment) , 厂商侧会部署服务端 (Host) 。他们之间的消息通信方式有4种: E(Equipment)、

- 1、设备端上报消息给服务端, 服务端收到消息, 无需响应 (即可以不回客户端消息, 客户端只管发, 不校验返回消息) 。即E->H。
- 2、设备端发起向服务端发起客户端请求, 服务端收到消息后, 需要回复客户端消息。即E->H 后 H->E。
- 3、服务端主动向客户端发起请求, 客户端收到消息后, 返回响应消息给服务端。即H->E后, E-> H。
- 4、服务端主动给客户端发送消息, 客户端收到消息, 无需回应服务端。即H->E。(很少有这样的场景)

#### 3、甲方的secs的标准不一定是完全遵循secs-gem模式标准

使用是否标准, 取决于厂商secs协议的服务端供应商是否标准。通常服务端厂商是按照标准来的, 但由于secs定义的场景不能覆盖所有的半导体设备协议的非半导体设备, 还有大量的业务交互式通过secs服务端供应商和厂家联合规划出对应的通讯设计规范。所以, 设备厂商的开发文档来源不仅仅网站的协议文档, 还有一部分是来源于secs服务端厂商提供的具体业务的开发文档。

当然semi最开始定位的半导体设备, 一些半导体的核心设备的功能肯定已经支持了, 这个可以只需要按照协议来就能搞定。如果是类半导体行业使用sec如面板行业, 通常都会有一些定制交互。

#### 4、secs的厂商测试流程

通常会分为:

- 1、基本对接: 完成客户端和服务端连接, 完成网络接入。
- 2、单元测试: 首先会完成基本业务的消息交互测试, 异常场景的消息交互测试。
- 3、in-line测试: 真实跑料过程, 测试真实的消息交互是否正常。
- 4、on-line测试: 正式生产, 管控正式生产的每一颗产品都能溯源, 即上下游设备的通信, 物料的识别关联, 上报。

### 四、

#### 1、开发模式

secs协议的开发模式通常分两种:

- 1.完全自主开发。(开发周期长, 需要secs, hsms, gem的详细协议文档, 完全自主开发工作量, 保守估计也要半年的工作量)
  - 2.选用secs库供应商, 在secs库基础上做二次开发。(开发周期短, 基本不关心hsms, secs, gem协议详情, 这个开发周期预期是1人月)
- 具体的开发模式选择, 根据厂商自己的项目周期, 人员配置, 成本自行选择。

#### 2、测试资源

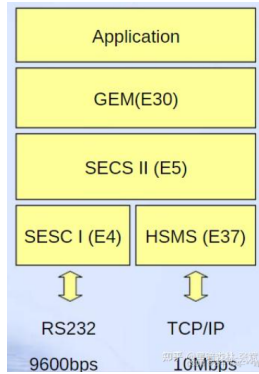
开发了一定要能测, 如何测也是一个问题。根据开发方式的不同, secs的测试的选择也会不同,

第一种: 自主开发会在测试端受阻, 但可以去准备一个服务端模拟器, 可以先期调通连



第二种：可以找供应商要服务端测试程序，一般都会提供。

## 五、与Host连接



从协议栈，我可以看到，连接方式有两种：

**串口连接：**对应的是secs i 协议。串口参数，串口号，波特率，数据位，停止位，校验位。

**网口连接：**对应的是HSMS协议。对应的参数是Host IP 和Port。

由于串口连接的方式现在已经很少用了，本文重点讲解网口连接方式，即HSMS连接。

### 1、准备工作

准备好secs协议模拟器（ITRI CIM Emulator）。

资源路径：ITRICIMEmulator模拟器.rar\_制造文档类资源-CSDN下载

准备好协议文档。

资源路径：Secs-II-E5-协议原文.zip\_互联网文档类资源-CSDN下载

### 2、基本概念扫盲

#### (1) 连接方式

**主动连接：**这个是设备软件在启动后，会主动与Host服务端进行连接。

此时配置

remote IP / port : Host IP/Port

local IP / port : Equipment IP/Port

**被动连接：**设备不主动与Host连接，而是等待Host发起连接。

此时只需要配置

remote IP / port : Equipment IP/Port

local IP / port : Equipment IP/Port

#### (2) 超时时间设置

T3 回复超时

指发送指令到接收到回复指令的最大时间

T5 连接间隔

指断开连接和重新连接的最小时间

T6 控制指令超时时间

主要指连接选择，取消选择，连接检测等控制指令的回复最大时间

T7 连接超时

指TCP/IP连接成功后到连接选择之间的最大时间，也就是发送stype=1 到收到stype=2 回复的这段时间

T8 接收超时

指接收到的两个字符之间的最大时间

(3) 在线, 离线, 连接, 未连接

**在线:** 设备与Host主机已对接成功, 且设备在host的服务端状态为在线, 可以接收Host命令。

**离线:** 设备在host服务端设置的状态是离线, 但连接是正常的。设备处于离线状态只响应S1F17和S1F13指令 其他指令都会返回设备离线响应

**连接:** 设备与Host主机已对接成功。

**未连接:** 设备与Host主机还未连接。

(4) 基本概念

DeviceID: 用于Host识别的设备号。

CEID:event ID 即事件ID

SVID:设备状态变量ID

ECID:设备常量ID

VID : 变量ID

RPTID: report ID 即上报ID

PPID: recipe ID 即配方ID (工艺ID)

CHIPID: 芯片ID

(5) 命令字

secs的核心设计就是命令交互。secs对命令字的功能都有详细的定义。

命令字由S和F组成, S-Stream F-Function 简单理解, S是一级目录, F是二级目录。

例如:

S1F5 命令字向请求设备状态。H->E

S1F6命令字响应S1F5, 上报设备状态。E->H

### 3、连接交互

(1) 通信确认存在

REQUEST: S1F1 are you there request

RESPONSE: S1F2 online data

(2) 确认状态

REQUEST:S1F3

RESPONSE:S1F4

(3) 建立连接

REQUEST:S1F13

RESPONSE:S1F14

(4) 离线通知

REQUEST:S1F15

RESPONSE:S1F16

(5) 上线通知

REQUEST:S1F17

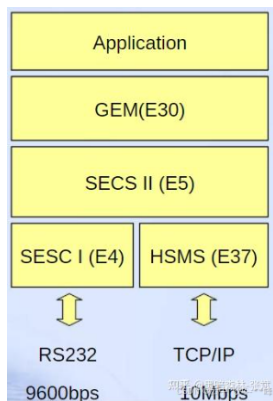
RESPONSE:S1F18

### 六、SML格式讲解



云中飞鸿--\*\*峰

已关注



这里我们不讨论secs i和HSMS的格式，我们只讨论secsii的协议，因为secs ii的报文内容的格式就是sml格式。

secs ii协议的消息报文分为两部分：消息命令字 + 消息体。其中消息体采用的是SML格式，本文对SML格式进行简单讲解，帮助大家了解SML的格式和后续的开发。

sml格式有些像xml，都是属于纯文本格式，但与xml又不同，他定义了它自身的语法。例如下面一段报文：

AI生成项目

```

1 // 命令字
2
3 S1F14:
4
5 // 消息体 sml 格式
6
7 <L[2]
8
9     <B[1] 0x00>
10
11     <L[0]>
12
13 >
  
```

SML以 <> 作为开始和结束。有自定义类型和元素个数。类型后面跟[]，中括号里表示的元素的个数。多个的情况下，是空格间隔即可。字符类型A的用双引号引起来。

元素类型：

AI生成项目

```

1 L: List    这个可以组织不同类型的元素在一起。可以嵌套List。
2
3 例: <L[3]    //表示3个元素,
4
5         <B[0] 0x00>
6
7         <U1[0]>
8
9         <L[2]>
10
11            <U4[0]>
12
13            <A[2] "ok">
14
15        >
16
17 B: binary    例: <B[2] 0x00 0x01>
18
19 U1: 1个字节的整形    例: <U1[2] 20 10>
20
21 U2: 2个字节的整形    例: <U2[2] 20 10>
22
23 U4: 4个字节的整形例    例: <U4[2] 20 10>
24
25 A:  ASCII码字符    例: <A[5] "hello">
26
  
```

 云中飞鸿--\*\*峰 已关注

### 七、交互测试

300mm晶圆是目前IC制造的主流，国际半导体装备和材料协会（SEMI）及全球半导体制造厂、设备供应商共同起草并制定了一系列300mm相关的自信标准，规定了生产管理系统（MES/CIM）与IC装备之间的通信协议。

目前，与通信相关的SEMI软件标准有如下3类：

**第一类：**SECS/GEM标准：传统的主机设备通信标准，定义了通信的物理层（RS232, TCP/IP）、消息格式以及设备通信场景。

**第二类：**300mm相关标准：以SECS/GEM标准为基础，针对300mm晶圆加工过程中的特殊需求，定义了加工过程管理、批量物料的作业管理、物料管理、载体并行I/O接口等。

**第三类：**Interface A标准：Interface A通信方法工具软件是设备系统的一部分，由Interface A接口、服务和数据类型，以及设备模型工具和设备控制系



基础库-设备端（C++版）：

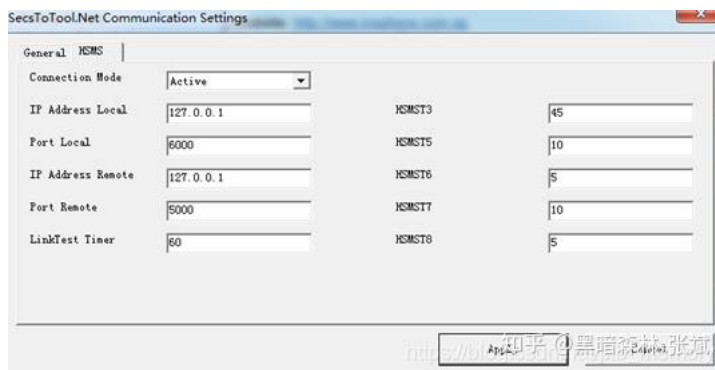
随着多个项目的实施，用户的需求也不断的增加，尤其是对设备端功能的需求日渐增加，因此开发库的设备端功能也得到了不断的完善，C++版本的：支持主机端又支持设备端的，此处演示设备端应用。由于设备端不像主机端一样有用户界面，因此为了说明可用性采取与第三方测试软件（“Swift Emr 信的方式”。

测试过程如下：

1. 首先启动第三方工具软件“Swift Emulator.exe”，点击“File”->“Open”，在弹出对话框中选择“Host.xml”，此时工具软件是作为主机端，如果作为设备：“Equipment.xml”；
2. 然后选择“Commuicaion”->“Settings”，弹出配置对话框，有“General”和“HSMS”两个标签；
3. “General”标签，主要是配置设备ID和通信协议，其中设备ID一定要与实际设备的ID一致，否则无法建立通信（默认为1），此时需要把设备ID修改为0，因为设备端软件默认ID为0，如下图所示；



4. “HSMS”标签，是配置通信地址、端口号和超时信息（保持默认即可）等，如下图所示：



以上配置暂时不用修改，保持默认即可，此时主机端为主动模式“Active”；

5. 然后选择“Commnuicaion“->“Open Port”，软件进入监听状态；

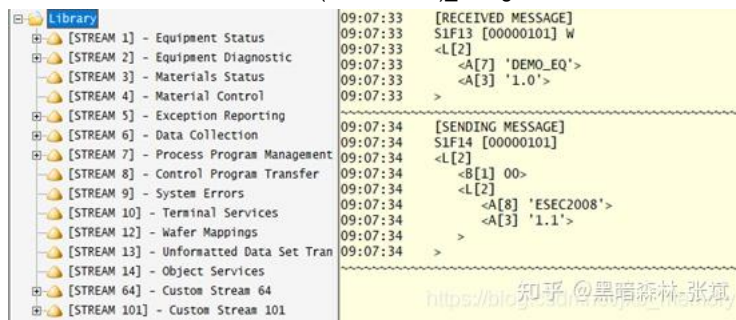
6. 启动设备端软件“DEMOSecs.exe”，然后输入“E”（选择了作为设备），再输入”P”（选择了作为被动模式）。此处如果选择“H”代表的是作为主机，；的是主动模式。

```
Choose mode, Host or Equipment? <Enter H or E>:
E
Choose HSMS mode, Passive or Active? <Enter P or A>:
P
```

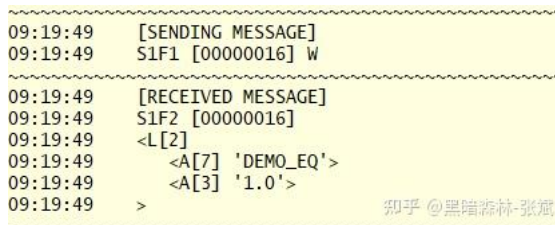
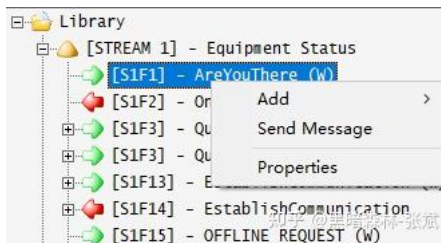
7. 回车后，设备端软件将等待主机端主动建立连接，主机端软件主动和设备端软件建立连接后，消息显示如下图所示：

```
Choose mode, Host or Equipment? (Enter H or E):
E
Choose HSMS mode, Passive or Active? (Enter P or A):
P
>>> Trial mode: service will stop after 20 minutes. <<<
SIM equipment started, PASSIVE mode.
Test Command List: (Input command character and press <RETURN>)
*****
* h:  * Print command list. *
* q:  * Exit. *
* i:  * List current info. *
* 1:  Send alarm to host.
* 2:  Send event to host.
* 3:  Send s14f1 to equipment.
* 4:  Send s14f2 to host.
* 5:  close secs service.
* 6:  restart service.
* 7:  Connect state.
*****
establish communication timeout expired.
establish communication timeout expired.
establish communication timeout expired.
establish communication timeout expired.
Host connected
establish communication timeout expired.
*****
2020-04-09 09:10:40.508
[S1F14] [MessageId 101]
<L
<B 00 >
<L
  <A ESEC2008>
  <A 1.1>
>
.>
*****
```

此处打印出了测试用的命令列表，可以输入进行测试。



8. 此时选择发送“S1F1”消息，则设备端会恢复“S1F2”消息，如下图所示；



9. 在设备端命令行界面输入“1”回车，设备将发送一个报警（S5F1）消息到主机端，如下图所示：



10. 目前设备端软件仅仅是一个Demo，支持的命令很有限，需要用户自行开发所要支持的命令。其实在使用此C++库开发Demo的过程本身就是在开能，只是由于是DEMO的原因支持的命令比较少，但实际上可以提供对所有SxFx系列消息的支持。

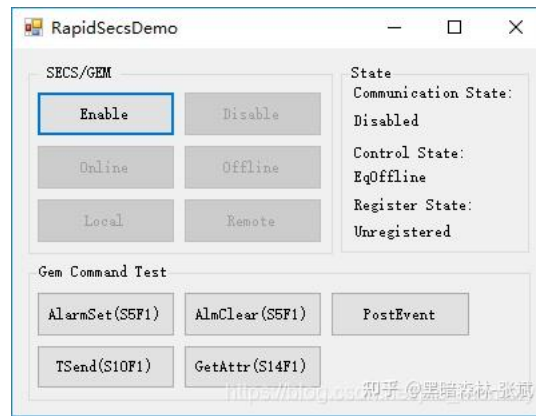
扩展库-设备端（C#版）：

C++版本的基础开发库由于没有提供对GEM（E30）的支持，需要用户自行开发，开发的工作量较大而且很多用户的开发能力有限、项目周期有限，提供对GEM的支持就成为了必须。

C#版扩展库的测试基本同C++版本一致，只不过可以进行测试的命令更多，但是要强调一点使用此扩展库也并不是不需要进行任何的二次开发（没有是可以不进行二次开发就可以直接使用的），都需要根据自己设备的实际情况进行相应的适配工作，比如设备SV、DV、EC的配置，还有各类事件、等等，本DEMO测试的前提是已经对这些内容进行了模拟适配。

测试过程如下：

1. 启动第三方工具软件“Swift Emulator.exe”，过程与上述相同就不再重复介绍了；
2. 启动设备端DEMO程序“RapidSecsDemo.exe”，界面如下：



窗口中有3个部分:

SECS/GEM: 此部分的按钮功能用于设备在通讯/控制状态机的各个状态之间进行切换(状态机的具体内容后面会讲);

State: 此部分显示的是设备当前的状态;

Gem Command Test: 此部分用于模拟测试设备主动向主机端发送的命令。

3. 点击“Enable”按钮后设备端Demo程序就进入了服务状态可以与主机端进行通信行了, 主机端与设备端建立连接后消息显示如下:

```

10:25:28 [SENDING MESSAGE]
10:25:28 S1F13 [00000054] W
10:25:28 <L[0]
10:25:28 >
~~~~~
10:25:28 [RECEIVED MESSAGE]
10:25:28 S1F14 [00000054]
10:25:28 <L[2]
10:25:28 <B[1] 00>
10:25:28 <L[2]
10:25:28 <A[12] 'SL_SIM_EQ111'>
10:25:28 <A[5] 'V1.00'>
10:25:28 >
~~~~~
10:25:28 [RECEIVED MESSAGE]
10:25:28 S1F1 [00000101] W
10:25:28 <L[2]
10:25:28 <A[12] 'SL_SIM_EQ111'>
10:25:28 <A[5] 'V1.00'>
10:25:28 >
~~~~~
10:25:28 [SENDING MESSAGE]
10:25:28 S1F2 [00000101]
~~~~~
10:25:28 [RECEIVED MESSAGE]
10:25:28 S6F11 [00000102] W
10:25:28 <L[3]
10:25:28 <U2[1] 0>
10:25:28 <U4[1] 9>
10:25:28 <L[0]
10:25:28 >
~~~~~
10:25:28 [SENDING MESSAGE]
10:25:28 S6F12 [00000102]
10:25:28 <B[1] 00>

```

之所以此处的消息比C++版的多是因为C#版本的扩展库支持了GEM, 并且此Demo配置了事件上报(S6F11), 所以此处出现了事件消息上报主机。

4. 后面在讲述Demo开发的时候将进行逐条命令的测试, 此处不进行详细的测试。本设备端Demo程序支持的命令列表如下:



云中飞鸿--\*\*峰

已关注

	命令	功能名 (符号)	方向
1	S1F1	Are You There Request(R)	设备<->主机
2	S1F2	On Line Data (D)	设备<->主机
3	S1F3	Selected Equipment Status Request (SSR)	主机->设备
4	S1F4	Selected Equipment Status Data (SSD)	设备->主机
5	S1F11	Status Variable Namelist Request (SVNR)	主机->设备
6	S1F12	Status Variable Namelist Reply (SVNRR)	设备->主机
7	S1F13	Establish Communication Request (CR)	设备<->主机
8	S1F14	Establish Communication Acknowledge (CRA)	设备<->主机
9	S1F15	Request OFF-LINE (ROFL)	主机->设备
10	S1F16	OFF-LINE Acknowledge (OFLA)	设备<->主机
11	S1F17	Request ON-LINE (RONL)	主机->设备
12	S1F18	ON-LINE Acknowledge (ONLA)	设备<->主机
13	S2F13	Equipment Constant Request (ECR)	主机->设备
14	S2F14	Equipment Constant Data (ECD)	设备->主机
15	S2F15	New Equipment Constant Send (ECS)	主机->设备
16	S2F16	New Equipment Constant Acknowledge (ECA)	设备->主机
17	S2F17	Date and Time Request (DTR)	设备<->主机
18	S2F18	Date and Time Data (DTD)	设备<->主机
19	S2F23	Trace Initialize Send (TIS)	主机->设备
20	S2F24	Trace Initialize Acknowledge (TIA)	设备->主机
21	S2F29	Equipment Constant Namelist Request (ECNR)	主机->设备
22	S2F30	Equipment Constant Namelist (ECN)	设备->主机
23	S2F31	Date and Time Set Request (DTS)	主机->设备
24	S2F32	Date and Time Set Acknowledge (DTA)	设备->主机
25	S2F33	Define Report (DR)	主机->设备
26	S2F34	Define Report Acknowledge (DRA)	设备->主机
27	S2F35	Link Event Report (LER)	主机->设备
28	S2F36	Link Event Report Acknowledge (LERA)	设备->主机
29	S2F37	Enable/Disable Event Report (EDER)	主机->设备
30	S2F38	Enable/Disable Event Report (EERA)	设备->主机
31	S2F41	Host Command Send (HCS)	主机->设备
32	S2F42	Host Command Acknowledge (HCA)	设备->主机
33	S2F49	Enhanced Remote Command	主机->设备
34	S2F50	Enhanced Remote Command Acknowledge	设备->主机
35	S5F1	Alarm Report Send (ARS)	设备->主机
36	S5F2	Alarm Report Acknowledge (ARA)	主机->设备
37	S5F3	Enable/Disable Alarm Send (EAS)	主机->设备
38	S5F4	Enable/Disable Alarm Acknowledge (EAA)	主机->设备
39	S5F5	List Alarms Request (LAR)	主机->设备
40	S5F6	List Alarm Data (LAD)	设备->主机
41	S5F7	List Enabled Alarm Request (LEAR)	主机->设备
42	S5F8	List Enabled Alarm Data (LEAD)	设备->主机
43	S6F1	Trace Data Send (TDS)	设备->主机
44	S6F2	Trace Data Acknowledge (TDA)	主机->设备
45	S6F11	Event Report Send (ERS)	设备->主机
46	S6F12	Event Report Acknowledge (ERA)	主机->设备
47	S6F15	Event Report Request (ERR)	主机->设备
48	S6F16	Event Report Data (ERD)	设备->主机
49	S6F19	Individual Report Request (IRR)	主机->设备
50	S6F20	Individual Report Data (IRD)	设备->主机
51	S7F1	Process Program Load Inquire(PP	
52	S7F2	Process Program Load Grant(PPG	



云中飞鸿--\*\*峰

已关注

53	S7F5	Process Program Request(PPR)	主机<->设备
54	S7F6	Process Program Data(PPD)	设备<->主机
55	S7F17	Delete Process Program Send(DPS)	主机->设备
56	S7F18	Delete Process Program Acknowledge(DPA)	设备->主机
57	S7F19	Current EPPD Request(RER)	主机->设备
58	S7F20	Current EPPD Data(RED)	设备->主机
59	S7F25	Formatted Process Program Request(FPR)	主机<->设备
60	S7F26	Formatted Process Program Data(FPD)	设备<->主机
61	S10F1	Terminal Request (TRN)	设备->主机
62	S10F2	Terminal Request Acknowledge (TRA)	主机->设备
63	S10F3	Terminal Display, Single (VTN)	主机->设备
64	S10F4	Terminal Display, Single Acknowledge (VTA)	设备->主机
65	S10F9	Broadcast (BCN)	主机->设备
66	S10F10	Broadcast Acknowledge (BCA)	设备->主机

编辑于

### secs/gem介绍

SECS/GEM是由国际半导体设备与材料协会 (SEMI) 的会员一起构建的连接性标准。该标准应用于在设备和工厂管理系统间的通讯标准。介绍SECS/GEM指的是一组用于

### SECS/GEM实现 (一) 半导体通讯协议软件, C、C++使用介绍

SECS/GEM设计与开发介绍、原生态完美支持C、C++，设计高效率快速支持SECS/GEM通讯协议，同时无缝支持GEM300、AHMS搬运系统的OHT天车\AGV小车等

### SECS/GEM协议开发详解:与Host连接与通信标准

SEMI E4-0699:SEMI设备通信标准1消息传送(SECS-I) SEMI E5-0304:SEMI设备通信标准2消息内容(SECS-II) SEMI E30-1103:通信的一般模式和SEMI设备的控制(GEM) SEMI E

### Secs/Gem第一讲·总结精华版(基于secs4net项目的ChatGpt介绍)-CSDN...

SECS 报文用 SML 表示结构清晰,比如 S1F13 是主机发送 Establish Communication Request,设备回 S1F14 表示应答成功。Item 是报文的数据结构,像Item.A()是ASCII 类型

SECS (SEMI Equipment Communications Standard) 是半导体设备通信的标准协议,主要用于设备与主机(如MES、EAP)之间的通信。1.SECS (SEMI Equipment C

### YMS系统开发2-EAP自动化SECS/GEM协议详解

secs/gem详解

### SECS/GEM PLC数据采集上报\_如何用secs获取设备的数据

SECS/GEM PLC数据采集上报 通过搭建OPC UA服务器,SECS软件和OPC 服务器交互,实现PLC 数据无缝采集并上传到EAP 优点: 1.零代码开发 2 兼容性强(适配市面上所有的

### SECS/GEM 简介

SECS/GEM提供多种方式让Host从设备中收集数据或信息: 可以随时使用 S1F3 指令请求一组状态变量值。 可以使用 S2F13 随时请求一组设备常量值。 主机可以定义包含

### 半导体设备通信标准-secs/gem协议概要介绍及使用方法, 以及关键指令说明

SECS/GEM是半导体行业用于设备 (Equipment) 与上层控制系统 (Host, 如MES/CIM) 之间通信的标准协议, 由SEMI (国际半导体产业协会): 定义设备行为模型、状

### C#\_半导体行业-SECS/GEM协议

一、概念 SECS/GEM 标准 (制造设备通信和控制的通用模型) 指的是一组用于半导体行业“设备与设备”之间或“设备与工厂主机系统”之间通信的国际标准协议, 由国际

### SECS/GSM 设备状态数据\_s1f3

SECS/GSM 设备状态数据 本文记录了一次在开发RMS过程中遇到的S1F3通讯指令解析异常的情况。当尝试通过S1F3指令获取设备SVID状态数据时,设备返回的数据除了第

### SECS GEM 常用指令概略\_secs s1f0

离线:设备在host服务端设置的状态是离线,但连接是正常的。设备处于离线状态只响应S1F17和S1F13指令 其他指令都会返回设备离线响应 连接:设备与Host主机已对接成功

### SECS/GEM协议 (HSMS协议)

头中的字节从第0字节(传输的第一个字节)编号到第9字节(传输的最后一个字节)。: 16位无符号整数, 它占据了字节0和字节1(字节0是MSB, 1是LSB)。: 8位无符号整数,

### SECS/GEM协议 (SECS-I)

本文介绍了SECS/GEM协议构成, SECS-I协议的具体内容。

### Secs/Gem第七讲(基于secs4net项目的ChatGpt介绍)\_s2f37

触发设备上报初始状态(S6F11 或 S1F1响应) 这个流程要么你在启动时自动做一遍,要么写在“设备上线事件”的回调里。 三、SECS4NET 中的自动重连机制 SECS4NET 默认

### ...HSMS/SECS/GEM半导体通讯协议 SVID管理S1F3 连接PLC modbus opc 三...

一、SECS协议的重要性 SECS/GEM(GenericModelfor SEMI Equipment Control)是由国际半导体设备与材料协会(SEMI)制定的一套标准通信协议,专为半导体生产设备和工厂



### SECS/GEM介绍

weixin\_40020850的

介绍SECS/GEM的内容和用途

### SECS/GEM协议 (GEM)

weixin\_47750287的

Host Initiated S1F13/F14 Scenario 主机启动的S1F13/F14场景。不同设备的状态模型在某些领域(如通信)将是相同的,但在其他领域(如处理)可能会有所不同。Event Noti

### SECS/GEM通信协议的使用

利用半导体通信协议以及结合硬件实现了**半导体设备自动化**

### SECSII--GEM协议

这个是半导体行业**标准SECS**协议的讲解,不是详细协议

### SECS/GEM概念

semis

SECS/GEM用于半导体行业的设备和工厂之间**通讯交互**,SECS/GEM是半导体**标准协会共同制定的标准**。半导体协会semi由众多的半导体公司参与和制定相关**标准**,更多

### SECS/GEM半导体通讯开发

参考:www.secsgem.cn 1.介绍 .支持SEMI E4(SECS I)、E5(SECS II)、E30 (GEM)、E37 (HSMS) **通讯标准**。 .扩展支持SEMI E39、E40、E87、E90、E94、E116、E14:

### 通讯协议-SECS/GEM

weixin\_43811791的

SECS (SEMI Equipment Communication standard) 是由国际**半导体设备与材料协会SEMI**开发的**半导体设备通讯标准**。它分为物理部分和逻辑部分。

### SECS/GEM通信介绍

码农的

基本介绍 SECS/GEM指的是一组用于管理制造设备和工厂主机系统之间通信的**半导体行业标准**。消息层**标准SEMI E5 SECS-II**定义了一个通用的消息结构和一个包含许多**标**

### 半导体通讯标准EAP、SECS/GEM视频介绍

semis

GEM (或SECS / GEM) 是由**半导体设备材料倡议制定**的一组**连接标准**。这些**标准**用于定义**自动化设备与主机工厂网络**之间的通信,从而实现**智能工厂制造**。SECS是**半导**

### 半导体semi通讯的协议secs/gem

SECS代表**半导体设备通讯标准**,GEM代表**通用设备通讯**。SECS/GEM协议的主要目的是提供一种**标准化的通讯方式**,以便**半导体制造设备之间能够实现互联互通**。通过SE

关于我们 招贤纳士 商务合作 寻求报道 400-660-0108 kefu@csdn.net 在线客服 工作时间 8:30-22:00

公安备案号11010502030143 京ICP备19004658号 京网文〔2020〕1039-165号 经营性网站备案信息 北京互联网违法和不良信息举报中心 家长监护 网络110报警服务 中国互联网举报中心 Chrome商店下载 账号管理规范 版权与免责声明 版权申诉 出版物许可证 营业执照 ©1999-2025北京创新乐知网络技术有限公司



云中飞鸿--\*\*峰

博客等级 码龄9年

484 2377 3953 1894  
原创 点赞 收藏 粉丝

已关注

私信

### 创作助手

大纲生成/代码生成/文章润色...

深度思考 (R1)



### 分类专栏

- STL源码剖析(PJ版) 4篇
- 开源框架与第三方库 2篇
- Json 1篇
- XML 7篇

展开全部

上一篇: C++实现的GEM与SECS协议源代...

下一篇: 半导体量测检测

### 最新评论



云中飞鸿--\*\*峰

已关注

imx6ull网络模块: 为什么要修改PHY地址...

weixin\_46402712: 说半天没说为什么删除74lv595

cmake 交叉编译(linux平台编译arm)

禾294: 你好, 可以帮忙编译一下吗, 有1

VMware虚拟机下的ubuntu不能连网解决

high\_np: 看了这么多还得是你

VMware虚拟机下的ubuntu不能连网解决

皮蛋和蛋蛋: 有用, 谢谢 😊

套接字I/O模型: 重叠模型 (1.事件通知)

CSDN-Ada助手: 哇, 你的文章质量真不错, 值得学习! 不过这么高质量的文章, 还值 ...

### 大家在看

- 网易算法岗位-面试真题分析
- Java微服务架构设计模式详解 2030
- 磁盘性能IOPS (五) -数据库scheduler从cfq到deadline提升效果
- 最新版虚拟机VMware安装文件下载及使用指南 (超详细) 735
- Java基础教程 (210) 设计模式之结构型模式: 设计模式之美: 深度解析Java结构型模式的七大法宝

### 最新文章

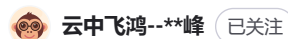
- C++17 新特性
- pair之于vector、queue (vector < pair < int,int > >)
- stringstream + getline()实现字符串分割

2025

08月 3篇	07月 15篇	06月 3篇	05月 2篇
04月 10篇	03月 13篇	02月 12篇	01月 14篇
2024年 253篇	2023年 162篇	2022年 2篇	2021年 9篇
2020年 32篇	2019年 134篇	2018年 16篇	

### 目录

- 前言
- 一、SECS/GEM 简介
- 二、系统的硬件组成
- 三、
  - 1、secs是一个协议族
  - 2、secs的消息通信
  - 3、甲方的secs的标准不一定是完全...
  - 4、secs的厂商测试流程
- 四、
  - 1、开发模式



2、测试资源

五 ESDS连接

收起 ^



云中飞鸿--\*\*峰

已关注