

# Module Interface Specification for STEM Moiré GPA

Alexandre Pofelski  
macid: pofelska  
github: slimpotatoes

November 16, 2017

# 1 Revision History

Date	Version	Notes
14/11/2017	1.0	First draft

## 2 Symbols, Abbreviations and Acronyms

The same Symbols, Abbreviations and Acronyms as in the SRS, the TestPlan and the MG (available in [STEM Moiré GPA](#) repository) are used in the Module Interface Specifications document.

addition to document

# Contents

<b>1</b>	<b>Revision History</b>	<b>i</b>
<b>2</b>	<b>Symbols, Abbreviations and Acronyms</b>	<b>ii</b>
<b>3</b>	<b>Introduction</b>	<b>1</b>
<b>4</b>	<b>Notation</b>	<b>1</b>
<b>5</b>	<b>Module Decomposition</b>	<b>1</b>
<b>6</b>	<b>MIS of Hardware Hiding Module (M 1)</b>	<b>3</b>
6.1	Module . . . . .	3
6.2	Uses . . . . .	3
6.3	Syntax . . . . .	3
6.3.1	Exported Access Programs . . . . .	3
6.4	Semantics . . . . .	3
6.4.1	State Variables . . . . .	3
6.4.2	Access Routine Semantics . . . . .	3
<b>7</b>	<b>MIS of STEM Moiré GPA Control Module (M 2)</b>	<b>3</b>
7.1	Module . . . . .	3
7.2	Uses . . . . .	3
7.3	Syntax . . . . .	4
7.3.1	Exported Access Programs . . . . .	4
7.4	Semantics . . . . .	4
7.4.1	State Variables . . . . .	4
7.4.2	Access Routine Semantics . . . . .	4
<b>8</b>	<b>MIS of STEM Moiré GPA GUI Module (M 3)</b>	<b>4</b>
8.1	Module . . . . .	4
8.2	Uses . . . . .	4
8.3	Syntax . . . . .	4
8.3.1	Exported Access Programs . . . . .	4
8.4	Semantics . . . . .	5
8.4.1	State Variables . . . . .	5
8.4.2	Access Routine Semantics . . . . .	5
<b>9</b>	<b>MIS of Input Module (M 4)</b>	<b>5</b>
9.1	Module . . . . .	5
9.2	Uses . . . . .	5
9.3	Syntax . . . . .	5
9.3.1	Exported Access Programs . . . . .	5

9.4	Semantics . . . . .	5
9.4.1	State Variables . . . . .	5
9.4.2	Access Routine Semantics . . . . .	5
<b>10</b>	<b>MIS of SMH Simulation (M 5)</b>	<b>6</b>
10.1	Module . . . . .	6
10.2	Uses . . . . .	6
10.3	Syntax . . . . .	6
10.3.1	Exported Access Programs . . . . .	6
10.4	Semantics . . . . .	6
10.4.1	State Variables . . . . .	6
10.4.2	Access Routine Semantics . . . . .	6
<b>11</b>	<b>MIS of GPA Module (M 6)</b>	<b>6</b>
11.1	Module . . . . .	6
11.2	Uses . . . . .	7
11.3	Syntax . . . . .	7
11.3.1	Exported Access Programs . . . . .	7
11.4	Semantics . . . . .	7
11.4.1	State Variables . . . . .	7
11.4.2	Access Routine Semantics . . . . .	7
<b>12</b>	<b>MIS of Mask Module (M 7)</b>	<b>7</b>
12.1	Module . . . . .	7
12.2	Uses . . . . .	7
12.3	Syntax . . . . .	8
12.3.1	Exported Access Programs . . . . .	8
12.4	Semantics . . . . .	8
12.4.1	State Variables . . . . .	8
12.4.2	Access Routine Semantics . . . . .	8
<b>13</b>	<b>MIS of Unstrained region (M 8)</b>	<b>8</b>
13.1	Module . . . . .	8
13.2	Uses . . . . .	8
13.3	Syntax . . . . .	8
13.3.1	Exported Access Programs . . . . .	8
13.4	Semantics . . . . .	9
13.4.1	State Variables . . . . .	9
13.4.2	Access Routine Semantics . . . . .	9
<b>14</b>	<b>MIS of Conversion Module (M 9)</b>	<b>9</b>
14.1	Module . . . . .	9
14.2	Uses . . . . .	9

14.3	Syntax . . . . .	9
14.3.1	Exported Access Programs . . . . .	9
14.4	Semantics . . . . .	9
14.4.1	State Variables . . . . .	9
14.4.2	Access Routine Semantics . . . . .	9
<b>15</b>	<b>MIS of 2D Strain Tensor Module (M 10)</b>	<b>10</b>
15.1	Module . . . . .	10
15.2	Uses . . . . .	10
15.3	Syntax . . . . .	10
15.3.1	Exported Access Programs . . . . .	10
15.4	Semantics . . . . .	10
15.4.1	State Variables . . . . .	10
15.4.2	Access Routine Semantics . . . . .	10
<b>16</b>	<b>MIS of Fourier Transform Module (M 11)</b>	<b>10</b>
16.1	Module . . . . .	10
16.2	Uses . . . . .	10
16.3	Syntax . . . . .	11
16.3.1	Exported Access Programs . . . . .	11
16.4	Semantics . . . . .	11
16.4.1	State Variables . . . . .	11
16.4.2	Access Routine Semantics . . . . .	11
<b>17</b>	<b>MIS of Gradient Module (M 12)</b>	<b>11</b>
17.1	Module . . . . .	11
17.2	Uses . . . . .	12
17.3	Syntax . . . . .	12
17.3.1	Exported Access Programs . . . . .	12
17.4	Semantics . . . . .	12
17.4.1	State Variables . . . . .	12
17.4.2	Access Routine Semantics . . . . .	12
<b>18</b>	<b>MIS of Least Square Fit Method Module (M 13)</b>	<b>12</b>
18.1	Module . . . . .	12
18.2	Uses . . . . .	12
18.3	Syntax . . . . .	12
18.3.1	Exported Access Programs . . . . .	12
18.4	Semantics . . . . .	13
18.4.1	State Variables . . . . .	13
18.4.2	Access Routine Semantics . . . . .	13

<b>19 MIS of Phase Operation Module (M 14)</b>	<b>13</b>
19.1 Module . . . . .	13
19.2 Uses . . . . .	13
19.3 Syntax . . . . .	13
19.3.1 Exported Access Programs . . . . .	13
19.4 Semantics . . . . .	13
19.4.1 State Variables . . . . .	13
19.4.2 Access Routine Semantics . . . . .	13
<b>20 MIS of Data Structure Module (M 15)</b>	<b>14</b>
20.1 Module . . . . .	14
20.2 Uses . . . . .	14
20.3 Syntax . . . . .	14
20.3.1 Exported Access Programs . . . . .	14
20.4 Semantics . . . . .	14
20.4.1 State Variables . . . . .	14
20.4.2 Access Routine Semantics . . . . .	14
<b>21 MIS of Generic GUI/Plot Module (M 16)</b>	<b>15</b>
21.1 Module . . . . .	15
21.2 Uses . . . . .	15
21.3 Syntax . . . . .	15
21.3.1 Exported Access Programs . . . . .	15
21.4 Semantics . . . . .	15
21.4.1 State Variables . . . . .	15
21.4.2 Access Routine Semantics . . . . .	15
<b>22 Appendix</b>	<b>16</b>

### 3 Introduction

The following document details the Module Interface Specifications for STEM Moiré GPA. The full documentation and implementation can be found in [STEM Moiré GPA](#) repository.

### 4 Notation

The structure of the MIS for modules comes from [?], with the addition that template modules have been adapted from [?]. The mathematical notation comes from Chapter 3 of [?]. For instance, the symbol  $:=$  is used for a multiple assignment statement and conditional rules follow the form  $(c_1 \Rightarrow r_1 | c_2 \Rightarrow r_2 | \dots | c_n \Rightarrow r_n)$ .

The following table summarizes the primitive data types used by STEM Moiré GPA.

Data Type	Notation	Description
character	char	a single symbol or digit
integer	$\mathbb{Z}$	an integer number
natural number	$\mathbb{N}$	a natural number
real	$\mathbb{R}$	a real number

The specification of STEM Moiré GPA uses some derived data types: sequences, strings, and tuples. Sequences are lists filled with elements of the same data type. Strings are sequences of characters. Tuples contain a list of values, potentially of different types. In addition, STEM Moiré GPA uses functions, which are defined by the data types of their inputs and outputs. Local functions are described by giving their type signature followed by their specification.

### 5 Module Decomposition

The following table is taken directly from the Module Guide document for this project.



Level 1	Level 2
Hardware-Hiding Module	
	Input
	STEM Moiré GPA Control
	STEM Moiré GPA GUI
	User Input
Behaviour-Hiding Module	SMH simulation
	GPA
	Mask
	Unstrained region
	Conversion
	2D strain tensor
	Fourier Transform
	Least square fitting method
	Phase calculation
Software Decision Module	Gradient
	Generic GUI/Plot
	Data structure
	Object structure

Table 1: Module Hierarchy

LIST ALL MIS to refer them in other document

## 6 MIS of Hardware Hiding Module (M 1)

### 6.1 Module

### 6.2 Uses

Data Structure

### 6.3 Syntax

#### 6.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

### 6.4 Semantics

#### 6.4.1 State Variables

#### 6.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

## 7 MIS of STEM Moiré GPA Control Module (M 2)

### 7.1 Module

main

### 7.2 Uses

- STEM Moiré GPA GUI
- Processing modules
  - Unstrained region
  - Conversion
  - SMH Simulation
  - GPA

- 2D Strain Tensors

- Data Structure

## 7.3 Syntax

### 7.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

## 7.4 Semantics

STEM Moiré GPA is designed to have the different steps of the process flow driven by user directly through GUI\_SMG. The STEM Moiré GPA Control Module uses the events in STEM Moiré GPA GUI to use the processing modules in the order defined by the user.

### 7.4.1 State Variables

### 7.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

## 8 MIS of STEM Moiré GPA GUI Module (M 3)

### 8.1 Module

GUI\_SMG

### 8.2 Uses

- Generic GUI/Plot
- Data Structure

## 8.3 Syntax

### 8.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

## 8.4 Semantics

STEM Moiré GPA process flow is driven by user through GUI\_SMG. User triggers the events that start the wished processing step.

### 8.4.1 State Variables

### 8.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

## 9 MIS of Input Module (M 4)

### 9.1 Module

Input

### 9.2 Uses

- STEM Moiré GPA GUI
- Data Structure

### 9.3 Syntax

#### 9.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

## 9.4 Semantics

### 9.4.1 State Variables

### 9.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

## 10 MIS of SMH Simulation (M 5)

### 10.1 Module

SMHSim

### 10.2 Uses

- Fourier Transform
- Input
- Data Structure

### 10.3 Syntax

#### 10.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

### 10.4 Semantics

#### 10.4.1 State Variables

#### 10.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

## 11 MIS of GPA Module (M 6)

### 11.1 Module

GPA

## 11.2 Uses

- Mask
- Fourier Transform
- Phase
- Gradient
- Data Structure

## 11.3 Syntax

### 11.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

## 11.4 Semantics

### 11.4.1 State Variables

### 11.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

# 12 MIS of Mask Module (M 7)

## 12.1 Module

GPA

## 12.2 Uses

- Mask
- Fourier Transform
- Phase
- Gradient
- Data Structure

## 12.3 Syntax

### 12.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

## 12.4 Semantics

### 12.4.1 State Variables

### 12.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

## 13 MIS of Unstrained region (M 8)

### 13.1 Module

URef

### 13.2 Uses

- Least Square Fit
- Input
- Data Structure

## 13.3 Syntax

### 13.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

## 13.4 Semantics

### 13.4.1 State Variables

### 13.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

## 14 MIS of Conversion Module (M 9)

### 14.1 Module

MtoCConv

### 14.2 Uses

- Input
- Data Structure

### 14.3 Syntax

#### 14.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

## 14.4 Semantics

### 14.4.1 State Variables

### 14.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:



## 15 MIS of 2D Strain Tensor Module (M 10)

### 15.1 Module

2D\_Strain

### 15.2 Uses

Data Structure

### 15.3 Syntax

#### 15.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

### 15.4 Semantics

#### 15.4.1 State Variables

#### 15.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

## 16 MIS of Fourier Transform Module (M 11)

*# 2D Fourier transform*

### 16.1 Module

FTCalc

### 16.2 Uses

Data Structure

## 16.3 Syntax

### 16.3.1 Exported Access Programs

Name	In	Out	Exceptions
$\mathcal{FT}$	$f : \mathbb{R}^2 \rightarrow \mathbb{R}$	$f : \mathbb{R}^2 \rightarrow \mathbb{C}$	-
$\text{i}\mathcal{FT}$	$f : \mathbb{R}^2 \rightarrow \mathbb{C}$	$f : \mathbb{R}^2 \rightarrow \mathbb{R}$	-

## 16.4 Semantics

### 16.4.1 State Variables

None

### 16.4.2 Access Routine Semantics

*# Calculate the 2D Fourier transform of a function  $f$*

$\mathcal{FT}(f(x, y))$ :

- output:  $\tilde{f}(\nu, \mu)$  such that

$$\forall(\nu, \mu) \in \mathbb{R}^2 \wedge \forall(x, y) \in \mathbb{R}^2, \tilde{f}(\nu, \mu) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2i\pi(\nu x + \mu y)} dx dy$$

- exception:

*# Calculate the 2D inverse Fourier transform of a function  $\tilde{f}$*

$\text{i}\mathcal{FT}(\tilde{f}(\nu, \mu))$ :

- output:  $f(x, y)$  such that

$$\forall(x, y) \in \mathbb{R}^2 \wedge \forall(\nu, \mu) \in \mathbb{R}^2, f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \tilde{f}(\nu, \mu) e^{2i\pi(\nu x + \mu y)} dx dy$$

- exception:

## 17 MIS of Gradient Module (M 12)

*# 2D Gradient*

### 17.1 Module

GradCalc

## 17.2 Uses

Data Structure

## 17.3 Syntax

### 17.3.1 Exported Access Programs

Name	In	Out	Exceptions
grad	$f : \mathbb{R}^2 \rightarrow \mathbb{R}$	$f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$	-

## 17.4 Semantics

### 17.4.1 State Variables

### 17.4.2 Access Routine Semantics

*# Calculate the 2D gradient of a 2D function f*

grad((f(x, y)):

- output:  $\nabla f(x, y)$  such that

$$\forall (x, y) \in \mathbb{R}^2, \nabla f(x, y) = \begin{bmatrix} \frac{\partial f}{\partial x}(x, y) \\ \frac{\partial f}{\partial y}(x, y) \end{bmatrix}$$

- exception:

## 18 MIS of Least Square Fit Method Module (M 13)

*# 2D linear least square method to fit a function f*

## 18.1 Module

LSFMCalc

## 18.2 Uses

Data Structure

## 18.3 Syntax

### 18.3.1 Exported Access Programs

Name	In	Out	Exceptions
lsfm	$f : \mathbb{R}^2 \rightarrow \mathbb{R}$	$f : \mathbb{R}^2 \rightarrow \mathbb{R}$	-

## 18.4 Semantics

### 18.4.1 State Variables

### 18.4.2 Access Routine Semantics

# Calculate the 2D fit of a function  $f$  using the linear least square method on a domain  $U = ([x_0, x_1]; [y_0, y_1]) \in \mathbb{R}^2$

lsfm(f,U):

- output:  $fit(x, y) = ax + by$  such that

$$\begin{aligned} \forall (x, y) \in U, E(a, b) = \int_{x_0}^{x_1} \int_{y_0}^{y_1} [f(x, y) - fit(x, y)]^2 dx dy \text{ is minimized} \\ \Rightarrow \frac{\partial E}{\partial a} = 0 \wedge \frac{\partial E}{\partial b} = 0 \Rightarrow a = \frac{\int_{x_0}^{x_1} \int_{y_0}^{y_1} x f(x, y) dx dy}{\int_{x_0}^{x_1} \int_{y_0}^{y_1} x^2 dx dy} \wedge b = \frac{\int_{x_0}^{x_1} \int_{y_0}^{y_1} y f(x, y) dx dy}{\int_{x_0}^{x_1} \int_{y_0}^{y_1} y^2 dx dy} \end{aligned}$$

- exception:

## 19 MIS of Phase Operation Module (M 14)

### 19.1 Module

PhaseCalc

### 19.2 Uses

Data Structure

### 19.3 Syntax

#### 19.3.1 Exported Access Programs

Name	In	Out	Exceptions
unwrap	$f : \mathbb{R}^2 \rightarrow [-\pi, \pi[$	$f : \mathbb{R}^2 \rightarrow \mathbb{R}$	-
wrap	$f : \mathbb{R}^2 \rightarrow \mathbb{R}$	$f : \mathbb{R}^2 \rightarrow [-\pi, \pi[$	-

## 19.4 Semantics

### 19.4.1 State Variables

### 19.4.2 Access Routine Semantics

wrap(f):

- output:  $g$  such that

$$\forall (x, y) \in \mathbb{R}^2, \exists k \in \mathbb{Z} | g(x, y) = f(x, y) + 2k\pi \wedge g(x, y) \in [-\pi, \pi[$$

- exception:

$\text{unwrap}(f)$ :

- output:  $g$  such that

$$\begin{aligned} & \forall (x, y) \in \mathbb{R}^2, \exists k \in \mathbb{Z} | g(x, y) = f(x, y) + 2k\pi \wedge g \text{ is continuous} \\ \Rightarrow & \forall (x, y) \in \mathbb{R}^2, \exists k \in \mathbb{Z} | \lim_{(x, y) \rightarrow (x_0, y_0)} g(x, y) = g(x_0, y_0) = f(x_0, y_0) + 2k\pi \end{aligned}$$

- exception:

## 20 MIS of Data Structure Module (M 15)

### 20.1 Module

DataStruct

### 20.2 Uses

### 20.3 Syntax

#### 20.3.1 Exported Access Programs

Name	In	Out	Exceptions
set	Metadata	-	-

### 20.4 Semantics

#### 20.4.1 State Variables

#### 20.4.2 Access Routine Semantics

$()$ :

- transition:
- output:
- exception:

## 21 MIS of Generic GUI/Plot Module (M 16)

### 21.1 Module

GUIGene

### 21.2 Uses

Hardware-Hiding Data Structure

### 21.3 Syntax

#### 21.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

### 21.4 Semantics

#### 21.4.1 State Variables

#### 21.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

## 22 Appendix