

**iAPBS Reference Manual**  
0.2.0

Generated by Doxygen 1.2.18

Thu Jun 22 09:22:53 2006



---

# Contents

<b>1</b>	<b>iAPBS File Index</b>	<b>1</b>
1.1	iAPBS File List . . . . .	1
<b>2</b>	<b>iAPBS File Documentation</b>	<b>3</b>
2.1	apbs_driver.c File Reference . . . . .	3
2.2	apbs_driver.h File Reference . . . . .	7
2.3	debug.c File Reference . . . . .	11
2.4	debug.h File Reference . . . . .	13



---

# Chapter 1

## iAPBS File Index

### 1.1 iAPBS File List

Here is a list of all documented files with brief descriptions:

<a href="#">apbs_driver.c</a> (The main iAPBS driver code) . . . . .	3
<a href="#">apbs_driver.h</a> (Header file for the main iAPBS driver) . . . . .	7
<a href="#">debug.c</a> (Routines for debugging the main code) . . . . .	11
<a href="#">debug.h</a> (Header file for debug helper functions) . . . . .	13



---

## Chapter 2

# iAPBS File Documentation

### 2.1 apbs\_driver.c File Reference

The main iAPBS driver code.

```
#include "apbs/apbs.h"
#include "apbs/nosh.h"
#include "apbs/mgparm.h"
#include "apbs/pbeparm.h"
#include "apbs/femparm.h"
#include "apbs/routines.h"
#include "apbs_driver.h"
#include "debug.h"
```

#### Functions

- int `apbsdrv_` (int \*nat, int \*ispara, int i\_pbeparm[16], int set\_pbeparm[13], double r\_pbeparm[7], double ionq[MAXION], double ionc[MAXION], double ionr[MAXION], int i\_mgparm[6], int set\_mgparm[9], int dime[3], int pdime[3], double grid[3], double glen[3], double center[3], double cglen[3], double fglen[3], double ccenter[3], double fcenter[3], double \*ofrac, double positionx[NATOMS], double positiony[NATOMS], double positionz[NATOMS], double radius[NATOMS], double acharge[NATOMS], int \*dbg, int ncalc[1], double esEnergy[NOSH\_MAXCALC], double npEnergy[NOSH\_MAXCALC], double apbsdx[NATOMS], double apbsdy[NATOMS], double apbsdz[NATOMS], double apbsqfx[NATOMS], double apbsqfy[NATOMS], double apbsqfz[NATOMS], double apbsibx[NATOMS], double apbsiby[NATOMS], double apbsibz[NATOMS], double apbsnpx[NATOMS], double apbsnpy[NATOMS], double apbsnpz[NATOMS], double apbsdbx[NATOMS], double apbsdby[NATOMS], double apbsdbz[NATOMS])

*Wrapper iAPBS function.*

- VPUBLIC int `iforceMG` (Vmem \*mem, NOsh \*nosh, PBEparm \*pbeparm, MGparm \*mgparm, Vpmg \*pmg, int \*nforce, AtomForce \*\*atomForce, Valist \*alist[NOSH\_MAXMOL], int debug)

*Calculate forces from MG solution.*

---

## 2.1.1 Detailed Description

The main iAPBS driver code.

**Author:**

Robert Konecny

**Note:**

Energy is returned in kJ/mol, forces in kJ/(mol/Å).

**Revision:**

1.17

**Id:**

apbs\_driver.c,v 1.17 2006/06/12 22:44:25 rok Exp

## 2.1.2 Function Documentation

**2.1.2.1** `int apbsdrv_(int * nat, int * ispara, int i_pbeparm[16], int set_pbeparm[13], double r_pbeparm[7], double ionq[MAXION], double ionc[MAXION], double ionr[MAXION], int i_mgparm[6], int set_mgparm[9], int dime[3], int pdime[3], double grid[3], double glen[3], double center[3], double cglen[3], double fglen[3], double ccenter[3], double fcenter[3], double * ofrac, double positionx[NATOMS], double positiony[NATOMS], double positionz[NATOMS], double radius[NATOMS], double acharge[NATOMS], int * dbg, int ncalc[1], double esEnergy[NOSH_MAXCALC], double npEnergy[NOSH_MAXCALC], double apbsdx[NATOMS], double apbsdy[NATOMS], double apbsdz[NATOMS], double apbsqfx[NATOMS], double apbsqfy[NATOMS], double apbsqfz[NATOMS], double apbsibx[NATOMS], double apbsiby[NATOMS], double apbsibz[NATOMS], double apbsnpx[NATOMS], double apbsnpy[NATOMS], double apbsnpz[NATOMS], double apbsdbx[NATOMS], double apbsdby[NATOMS], double apbsdbz[NATOMS])`

Wrapper iAPBS function.

**Author:**

Robert Konecny

**Parameters:**

*nat* Number of atoms  
*ispara* Is a parallel calculation (1) or is not (0)  
*i\_pbeparm* Generic PBE parameters (integer values)  
*set\_pbeparm* Set-switch for PBE parameters  
*r\_pbeparm* Generic PBE parameters (real values)  
*ionq* Mobile ion charge  
*ionc* Mobile ion concentration  
*ionr* Mobile ion radius  
*i\_mgparm* MG-specific parameters (integer values)  
*set\_mgparm* set-switch MG-specific parameters  
*dime* Grid dimensions  
*pdime* Grid of processors to be used in calculation

*grid* Grid spacing  
*glen* Grid side lengths  
*center* Grid center  
*cglen* Coarse grid side lengths  
*fglen* Fine grid side lengths  
*ccenter* Coarse grid center  
*fcenter* Fine grid center  
*ofrac* Overlap fraction between procs  
*positionx* Atomic coordinate (x)  
*positiony* Atomic coordinate (y)  
*positionz* Atomic coordinate (z)  
*radius* Atomic radii  
*acharge* Atomic charges  
*dbg* Debug verbosity flag  
*ncalc* The number of calculations in the calc array  
*esEnergy* Electrostatic energy  
*npEnergy* Non-polar energy  
*apbsdx* Total electrostatic force per atom (x coordinate)  
*apbsdy* Total electrostatic force per atom (y coordinate)  
*apbsdz* Total electrostatic force per atom (z coordinate)  
*apbsqfx* Fixed charge force (x)  
*apbsqfy* Fixed charge force (y)  
*apbsqfz* Fixed charge force (z)  
*apbsibx* Ionic boundary force (x)  
*apbsiby* Ionic boundary force (y)  
*apbsibz* Ionic boundary force (z)  
*apbsnpx* Non-polar force (x)  
*apbsnpy* Non-polar force (y)  
*apbsnpz* Non-polar force (z)  
*apbsdbx* Dielectric boundary force (x)  
*apbsdby* Dielectric boundary force (y)  
*apbsdbz* Dielectric boundary force (z)

**Returns:**

1 if successful, 0 otherwise

**2.1.2.2** **VPUBLIC** int iforceMG (Vmem \* *mem*, NOsh \* *nosh*, PBEparm \* *pbeparm*, MGparm \* *mgparm*, Vpmg \* *pmg*, int \* *nforce*, AtomForce \*\* *atomForce*, Valist \* *alist*[NOSH\_MAXMOL], int *debug*)

Calculate forces from MG solution.

**Note:**

Calculate and write out forces for MG calculation

**Parameters:**

*mem* Memory management

*nosh* stores input file information

*pbeparm* PBE parameters

*mgparm* MG parameters

*pmg* Vpmg object for calculation

*nforce* 0 => no forces, 1 => net forces, >1 => number of forces (1 per atom)

*atomForce* pointer to array of force objects

*alist* molecules

*debug* verbosity flag

**Returns:**

1 if successful, 0 otherwise

**Note:**

Sometimes (if nosh->bogus == 1) we just go through the motions, but don't assign any forces

**Author:**

Robert Konecny, based on forceMG

## 2.2 apbs\_driver.h File Reference

Header file for the main iAPBS driver.

### Functions

- VEXTERNC int [apbsdrv\\_](#) (int \*nat, int \*ispara, int i\_pbeparm[16], int set\_pbeparm[13], double r\_pbeparm[7], double ionq[MAXION], double ionc[MAXION], double ionr[MAXION], int i\_mgparm[6], int set\_mgparm[9], int dime[3], int pdime[3], double grid[3], double glen[3], double center[3], double cglen[3], double fglen[3], double ccenter[3], double fcenter[3], double \*ofrac, double positionx[NATOMS], double positiony[NATOMS], double positionz[NATOMS], double radius[NATOMS], double acharge[NATOMS], int \*dbg, int ncalc[1], double esEnergy[NOSH\_MAXCALC], double npEnergy[NOSH\_MAXCALC], double apbsdx[NATOMS], double apbsdy[NATOMS], double apbsdz[NATOMS], double apbsqfx[NATOMS], double apbsqfy[NATOMS], double apbsqfz[NATOMS], double apbsibx[NATOMS], double apbsiby[NATOMS], double apbsibz[NATOMS], double apbsnpx[NATOMS], double apbsnpy[NATOMS], double apbsnpz[NATOMS], double apbsdbx[NATOMS], double apbsdby[NATOMS], double apbsdbz[NATOMS])

*Wrapper iAPBS function.*

- VEXTERNC int [iforceMG](#) (Vmem \*mem, NOsh \*nosh, PBEparm \*pbeparm, MGparm \*mgparm, Vpmg \*pmg, int \*nforce, AtomForce \*\*atomForce, Valist \*alist[NOSH\_MAXMOL], int debug)

*Calculate forces from MG solution.*

### 2.2.1 Detailed Description

Header file for the main iAPBS driver.

**Author:**

Robert Konecny

**Revision:**

1.6

**Id:**

apbs\_driver.h,v 1.6 2006/03/31 00:31:16 rok Exp

## 2.2.2 Function Documentation

**2.2.2.1** `VEXTERNC int apbsdrv_ (int * nat, int * ispara, int i_pbeparm[16], int set_pbeparm[13], double r_pbeparm[7], double ionq[MAXION], double ionc[MAXION], double ionr[MAXION], int i_mgparm[6], int set_mgparm[9], int dime[3], int pdime[3], double grid[3], double glen[3], double center[3], double cglen[3], double fglen[3], double ccenter[3], double fcenter[3], double * ofrac, double positionx[NATOMS], double positiony[NATOMS], double positionz[NATOMS], double radius[NATOMS], double acharge[NATOMS], int * dbg, int ncalc[1], double esEnergy[NOSH_MAXCALC], double npEnergy[NOSH_MAXCALC], double apbsdx[NATOMS], double apbsdy[NATOMS], double apbsdz[NATOMS], double apbsqfx[NATOMS], double apbsqfy[NATOMS], double apbsqfz[NATOMS], double apbsibx[NATOMS], double apbsiby[NATOMS], double apbsibz[NATOMS], double apbsnpx[NATOMS], double apbsnpy[NATOMS], double apbsnpz[NATOMS], double apbsdbx[NATOMS], double apbsdby[NATOMS], double apbsdbz[NATOMS])`

Wrapper iAPBS function.

### Author:

Robert Konecny

### Parameters:

*nat* Number of atoms  
*ispara* Is a parallel calculation (1) or is not (0)  
*i\_pbeparm* Generic PBE parameters (integer values)  
*set\_pbeparm* Set-switch for PBE parameters  
*r\_pbeparm* Generic PBE parameters (real values)  
*ionq* Mobile ion charge  
*ionc* Mobile ion concentration  
*ionr* Mobile ion radius  
*i\_mgparm* MG-specific parameters (integer values)  
*set\_mgparm* set-switch MG-specific parameters  
*dime* Grid dimensions  
*pdime* Grid of processors to be used in calculation  
*grid* Grid spacing  
*glen* Grid side lengths  
*center* Grid center  
*cglen* Coarse grid side lengths  
*fglen* Fine grid side lengths  
*ccenter* Coarse grid center  
*fcenter* Fine grid center  
*ofrac* Overlap fraction between procs  
*positionx* Atomic coordinate (x)  
*positiony* Atomic coordinate (y)  
*positionz* Atomic coordinate (z)  
*radius* Atomic radii

*acharge* Atomic charges  
*dbg* Debug verbosity flag  
*ncalc* The number of calculations in the calc array  
*esEnergy* Electrostatic energy  
*npEnergy* Non-polar energy  
*apbsdx* Total electrostatic force per atom (x coordinate)  
*apbsdy* Total electrostatic force per atom (y coordinate)  
*apbsdz* Total electrostatic force per atom (z coordinate)  
*apbsqfx* Fixed charge force (x)  
*apbsqfy* Fixed charge force (y)  
*apbsqfz* Fixed charge force (z)  
*apbsibx* Ionic boundary force (x)  
*apbsiby* Ionic boundary force (y)  
*apbsibz* Ionic boundary force (z)  
*apbsnpx* Non-polar force (x)  
*apbsnpy* Non-polar force (y)  
*apbsnpz* Non-polar force (z)  
*apbsdbx* Dielectric boundary force (x)  
*apbsdby* Dielectric boundary force (y)  
*apbsdbz* Dielectric boundary force (z)

**Returns:**

1 if successful, 0 otherwise

**2.2.2.2 VEXTERNC** *int iforceMG (Vmem \* mem, NOsh \* nosh, PBEparm \* pbeparm, MGparm \* mgparm, Vpmg \* pmg, int \* nforce, AtomForce \*\* atomForce, Valist \* alist[NOSH\_MAXMOL], int debug)*

Calculate forces from MG solution.

**Note:**

Calculate and write out forces for MG calculation

**Parameters:**

*mem* Memory management  
*nosh* stores input file information  
*pbeparm* PBE parameters  
*mgparm* MG parameters  
*pmg* Vpmg object for calculation  
*nforce* 0 => no forces, 1 => net forces, >1 => number of forces (1 per atom)  
*atomForce* pointer to array of force objects  
*alist* molecules  
*debug* verbosity flag

**Returns:**

1 if successful, 0 otherwise

**Note:**

Sometimes (if nosh->bogus == 1) we just go through the motions, but don't assign any forces

**Author:**

Robert Konecny, based on forceMG

## 2.3 debug.c File Reference

Routines for debugging the main code.

```
#include "apbs/apbs.h"
#include "apbs/nosh.h"
#include "apbs/mgparm.h"
#include "apbs/pbeparm.h"
#include "apbs/femparm.h"
#include "apbs/routines.h"
#include "apbs_driver.h"
#include "debug.h"
```

### Functions

- void [pbe\\_check](#) (PBEparm \*pbeparm)  
*Checks pbeparm values.*
- void [mg\\_check](#) (MGparm \*mgparm)  
*Checks mgparm values.*
- void [nosh\\_check](#) (int i, NOsh \*nosh)  
*Check NOsh values.*

### 2.3.1 Detailed Description

Routines for debugging the main code.

**Author:**

Robert Konecny

**Revision:**

1.9

**Id:**

debug.c,v 1.9 2006/03/31 00:31:16 rok Exp

### 2.3.2 Function Documentation

#### 2.3.2.1 void mg\_check (MGparm \* mgparm)

Checks mgparm values.

**Author:**

Robert Konecny

**Parameters:**

*mgparm* MG-specific parameters

**2.3.2.2 void nosh\_check (int *i*, NOsh \* *nosh*)**

Check NOsh values.

**Author:**

Robert Konecny

**Parameters:**

*i* Calculation number

*nosh* Parameters from input file

**2.3.2.3 void pbe\_check (PBEparm \* *pbeparm*)**

Checks pbeparm values.

**Author:**

Robert Konecny

**Parameters:**

*pbeparm* Generic PBE parameters

## 2.4 debug.h File Reference

Header file for debug helper functions.

```
#include "apbs/apbs.h"
#include "apbs/nosh.h"
#include "apbs/mgparm.h"
#include "apbs/pbeparm.h"
#include "apbs/femparm.h"
#include "apbs/routines.h"
```

### Functions

- VEXTERNC void [pbe\\_check](#) (PBEparm \*pbeparm)  
*Checks pbeparm values.*
- VEXTERNC void [mg\\_check](#) (MGparm \*mgparm)  
*Checks mgparm values.*
- VEXTERNC void [alist\\_check](#) (Valist \*alist[0])  
*Checks alist values.*
- VEXTERNC void [nosh\\_check](#) (int i, NOsh \*nosh)  
*Check NOsh values.*

### 2.4.1 Detailed Description

Header file for debug helper functions.

**Author:**

Robert Konecny

**Revision:**

1.6

**Id:**

debug.h,v 1.6 2005/09/23 16:43:38 rok Exp

### 2.4.2 Function Documentation

#### 2.4.2.1 VEXTERNC void [alist\\_check](#) (Valist \* *alist*[0])

Checks alist values.

**Author:**

Robert Konecny

**Parameters:**

*alist* List of atom lists

**2.4.2.2 VEXTERNC void mg\_check (MGparm \* *mgparm*)**

Checks mgparm values.

**Author:**

Robert Konecny

**Parameters:**

*mgparm* MG-specific parameters

**2.4.2.3 VEXTERNC void nosh\_check (int *i*, NOsh \* *nosh*)**

Check NOsh values.

**Author:**

Robert Konecny

**Parameters:**

*i* Calculation number

*nosh* Parameters from input file

**2.4.2.4 VEXTERNC void pbe\_check (PBEparm \* *pbeparm*)**

Checks pbeparm values.

**Author:**

Robert Konecny

**Parameters:**

*pbeparm* Generic PBE parameters

---

# Index

alist\_check  
  debug.h, [13](#)

apbs\_driver.c, [3](#)  
  apbsdrv\_, [4](#)  
  iforceMG, [5](#)

apbs\_driver.h, [7](#)  
  apbsdrv\_, [8](#)  
  iforceMG, [9](#)

apbsdrv\_  
  apbs\_driver.c, [4](#)  
  apbs\_driver.h, [8](#)

debug.c, [11](#)  
  mg\_check, [11](#)  
  nosh\_check, [11](#)  
  pbe\_check, [12](#)

debug.h, [13](#)  
  alist\_check, [13](#)  
  mg\_check, [13](#)  
  nosh\_check, [14](#)  
  pbe\_check, [14](#)

iforceMG  
  apbs\_driver.c, [5](#)  
  apbs\_driver.h, [9](#)

mg\_check  
  debug.c, [11](#)  
  debug.h, [13](#)

nosh\_check  
  debug.c, [11](#)  
  debug.h, [14](#)

pbe\_check  
  debug.c, [12](#)  
  debug.h, [14](#)

---