AmigaMail

COLLABORATORS

	TITLE : AmigaMail	

ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		July 19, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

AmigaMail

1.1 I-35: Release 2.0 Specific Functions and Commands

Compiled by Carolyn Scheppner

The 2.0 Release of the Amiga Operating System contains hundreds of new library functions and device commands. The following list is a quick reference for all of the functions and commands that were introduced to the Amiga OS in releases 2.00 through 2.04 (V36 and V37). Only the 2.04 release contains all of these functions.

asl.library (basename: _AslBase) V36
battclock.resource (basename: _BattClockBase)
battmem.resource (basename: _BattMemBase)
clipboard.device (device commands)
commodities.library (basename: _CxBase) V36
disk.resource (basename: _DiskBase)
diskfont.library (basename: _DiskfontBase)
dos.library (basename: _DOSBase)
exec.library (basename: _SysBase)
expansion.library (basename: _ExpansionBase)
gadtools.library (basename: _GadToolsBase) V36
graphics.library (basename: _GfxBase)
icon.library (basename: _IconBase)
iffparse.library (basename: _IFFParseBase) V36
input.device (basename: _InputBase)
intuition.library (basename: _IntuitionBase)
keymap.library (basename: _KeymapBase)
layers.library (basename: _LayersBase)
mathieeesingbas.library (basename: _MathIeeeSingBasBase) V36
mathieeesingtrans.library (basename: _MathIeeeSingTransBase) V36
ramdrive.device (basename: _RamdriveDevice)
rexxsyslib.library (basename: _RexxSysBase) V36
Timer.Device (basename: _TimerBase)
trackdisk.device (device commands)
utility.library (basename: _UtilityBase) V36
workbench.library (basename: _WorkbenchBase)

1.2 asl.library (basename: _AslBase) V36

```
AllocAslRequest(type,tagList)(d0/a0)     - Also stack-based amiga.lib stub
                                           AllocAslRequestTags() alloc an
                                           ASL requester, with TagItem
                                           modifiers (V36)
AllocFileRequest()()                     - Allocates a FileRequester
                                           structure (V36)
AslRequest(request,tagList)(a0/a1)       - lso stack-based amiga.lib stub
                                           AslRequestTags().  Get input
                                           from user for an ASL requester
                                           (V36)
FreeAslRequest(request)(a0)              - Frees requester obtained from
                                           AllocAslRequest (V36)
FreeFileRequest(fileReq)(a0)             - Frees requester allocated by
                                           AllocFileRequest (V36)
RequestFile(fileReq)(a0)                 - request user to select file(s)
                                           (V36)
```

1.3 battclock.resource (basename: _BattClockBase)

```
ReadBattClock()()                        - Read time from clock chip.  (V36)

ResetBattClock()()                       - Reset the clock chip.  (V36)
WriteBattClock(time)(d0)                 - Set the time on the clock chip.
                                           (V36)
```

1.4 battmem.resource (basename: _BattMemBase)

```
ObtainBattSemaphore()()                  - Obtain access to nonvolatile
                                           ram. (V36)
ReadBattMem(buffer,offset,length)(a0,d0/d1)
                                         - Read a bitstring from
                                           nonvolatile ram.  (V36)
ReleaseBattSemaphore()()                 - Allow nonvolatile ram to others.
                                            (V36)
WriteBattMem(buffer,offset,length)(a0,d0/d1)
                                         - Write a bitstring to nonvolatile
                                           ram.  (V36)
```

1.5 clipboard.device (device commands)

```
CBD_CHANGEHOOK                           - Add or remove a clip change
                                           hook.
```

1.6 commodities.library (basename: _CxBase) V36

```
ActivateCxObj(co,true)(a0,d0)           - Change the activation state of a
                                          commodity object.
AddIEvents(events)(a0)                  - Add input events to commodities'
                                          input stream. (V36)
AttachCxObj(headobj,co)(a0/a1)          - Attach a commodity object to the
                                          end of an existing
ClearCxObjError(co)(a0)                 - Clear the accumulated error
                                          value of a commodity
CreateCxObj(type,arg1,arg2)(d0/a0/a1)   - Create a new commodity object.
                                          (V36)
CxBroker(nb,error)(a0,d0)               - Create a commodity broker. (V36)
CxMsgData(cxm)(a0)                      - Obtain a pointer to a commodity
                                          message's data area. (V36)
CxMsgID(cxm)(a0)                        - Obtain the ID of a commodity
                                          message. (V36)
CxMsgType(cxm)(a0)                      - Obtain the type of a commodity
                                          message. (V36)
CxObjError(co)(a0)                      - Obtain a commodity object's
                                          accumulated error. (V36)
CxObjType(co)(a0)                       - Obtain the type of a commodity
                                          object. (V36)
DeleteCxObj(co)(a0)                     - Delete a commmodity object. (V36)

DeleteCxObjAll(co)(a0)                  - Recursively delete a tree of
                                          commodity objects.
DisposeCxMsg(cxm)(a0)                   - Delete a commodity message. (V36)

DivertCxMsg(cxm,headobj,ret)(a0/a1/a2)  - Send a commodity message down an
                                          object list. (V36)
EnqueueCxObj(headobj,co)(a0/a1)         - Insert a commodity object within
                                          a list of objects
InsertCxObj(headobj,co,pred)(a0/a1/a2)  - Insert a commodity object in a
                                          list after a given
InvertKeyMap(ansicode,event,km)(d0/a0/a1)
                                        - Generate an input event from an
                                          ANSI code. (V36)
ParseIX(description,ix)(a0/a1)          - Initialize an input expression
                                          given a description string.
RemoveCxObj(co)(a0)                     - Remove a commodity object from a
                                          list. (V36)
RouteCxMsg(cxm,co)(a0/a1)               - Set the next destination of a
                                          commodity message. (V36)
SetCxObjPri(co,pri)(a0,d0)              - Set the priority of a commodity
                                          object. (V36)
SetFilter(filter,text)(a0/a1)           - Change the matching condition of
                                          a commodity filter.
SetFilterIX(filter,ix)(a0/a1)           - Change the matching condition of
                                          a commodity filter.
SetTranslate(translator,events)(a0/a1)  - Replace a translator object's
                                          translation list. (V36)
```

1.7 disk.resource (basename: _DiskBase)

```
ReadUnitID(unitNum)(d0)                 - Reread and return the type of
                                          drive (V37)
```

1.8 diskfont.library (basename: _DiskfontBase)

```
NewScaledDiskFont(sourceFont,destTextAttr)(a0/a1)
                                        - Create a DiskFont scaled from
                                          another. (V36)
```

1.9 dos.library (basename: _DOSBase)

```
AbortPkt(port,pkt)(d1/d2)               - Aborts an asynchronous packet,
                                          if possible. (V36)
AddBuffers(name,number)(d1/d2)          - Changes the number of buffers
                                          for a filesystem (V36)
AddDosEntry(dlist)(d1)                  - Add a Dos List entry to the
                                          lists (V36)
AddPart(dirname,filename,size)(d1/d2/d3)- Appends a file/dir to the end of
                                          a path (V36)
AddSegment(name,seg,system)(d1/d2/d3)   - Adds a resident segment to the
                                          resident list (V36)
AllocDosObject(type,tags)(d1/d2)        - Creates a dos object (V36)
AssignAdd(name,lock)(d1/d2)             - Adds a lock to an assign for
                                          multi-directory assigns (V36)
AssignLate(name,path)(d1/d2)            - Creates an assignment to a
                                          specified path later (V36)
AssignLock(name,lock)(d1/d2)            - Creates an assignment to a
                                          locked object (V36)
AssignPath(name,path)(d1/d2)            - Creates an assignment to a
                                          specified path (V36)
AttemptLockDosList(flags)(d1)           - Attempt to lock the Dos Lists
                                          for use (V36)
ChangeMode(type,fh,newmode)(d1/d2/d3)   - Change the current mode of a
                                          lock or filehandle (V36)
CheckSignal(mask)(d1)                   - Checks for break signals (V36)
Cli()()                                 - Returns a pointer to the CLI
                                          structure of the process (V36)
CliInitNewcli(dp)(a0)                   - Set up a process as a shell
                                          according to the inital packet.
CliInitRun(dp)(a0)                      - Set up a process as a shell
                                          according to the inital packet.
CompareDates(date1,date2)(d1/d2)        - Compares two datestamps (V36)
CreateNewProc(tags)(d1)                 - Create a new process (V36)
DateToStr(datetime)(d1)                 - Converts a DateStamp to a string
                                          (V36)
DeleteVar(name,flags)(d1/d2)            - Deletes a local or environment
                                          variable (V36)
DoPkt(port,action,arg1,arg2,arg3,arg4,arg5)(d1/d2/d3/d4/d5/d6/d7)
                                        - Send a dos packet and wait for
                                          reply (V36)
```

```
DupLockFromFH(fh)(d1)                    - Gets a lock on an open file (V36)

EndNotify(notify)(d1)                    - Ends a notification request (V36)

ErrorReport(code,type,arg1,device)(d1/d2/d3/d4)
                                         - Displays a Retry/Cancel
                                           requester for an error (V36)
ExAll(lock,buffer,size,data,control)(d1/d2/d3/d4/d5)
                                         - Examine an entire directory (V36)

ExamineFH(fh,fib)(d1/d2)                 - Gets information on an open file
                                           (V36)
Fault(code,header,buffer,len)(d1/d2/d3/d4)
                                         - Returns the text associated with
                                           a DOS error code (V36)
FGetC(fh)(d1)                            - Read a character from the
                                           specified input (buffered) (V36)
FGets(fh,buf,buflen)(d1/d2/d3)           - Reads a line from the specified
                                           input (buffered) (V36)
FilePart(path)(d1)                       - Returns the last component of a
                                           path (V36)
FindArg(keyword,template)(d1/d2)         - Find a keyword in a template
                                           (V36)
FindCliProc(num)(d1)                     - Returns a pointer to the
                                           requested CLI process (V36)
FindDosEntry(dlist,name,flags)(d1/d2/d3)- Finds a specific Dos List entry
                                           (V36)
FindSegment(name,seg,system)(d1/d2/d3)   - Finds a segment on the resident
                                           list (V36)
FindVar(name,type)(d1/d2)                - Finds a local variable (V36)
Flush(fh)(d1)                            - Flushes buffers for a buffered
                                           filehandle (V36)
Format(filesystem,volumename,dostype)(d1/d2/d3)
                                         - Causes a filesystem to
                                           initialize itself (V36)
FPutC(fh,ch)(d1/d2)                      - Write a character to the
                                           specified output (buffered) (V36)
FPuts(fh,str)(d1/d2)                     - Writes a string the the
                                           specified output (buffered) (V36)
FRead(fh,block,blocklen,number)(d1/d2/d3/d4)
                                         - Reads a number of blocks from an
                                           input (buffered) (V36)
FreeArgs(args)(d1)                       - Free allocated memory after
                                           ReadArgs() (V36)
FreeDeviceProc(dp)(d1)                   - Releases port returned by
                                           GetDeviceProc() (V36)
FreeDosEntry(dlist)(d1)                  - Frees an entry created by
                                           MakeDosEntry (V36)
FreeDosObject(type,ptr)(d1/d2)           - Frees an object allocated by
                                           AllocDosObject() (V36)
FWrite(fh,block,blocklen,number)(d1/d2/d3/d4)
                                         - Writes a number of blocks to an
                                           output (buffered) (V36)
GetArgStr()()                            - Returns the arguments for the
                                           process (V36)
GetConsoleTask()()                       - Returns the default console for
                                           the process (V36)
```

```
GetCurrentDirName(buf,len)(d1/d2)       - Returns the current directory
                                          name (V36)
GetDeviceProc(name,dp)(d1/d2)           - Finds a handler to send a
                                          message to (V36)
GetFileSysTask()()                      - Returns the default filesystem
                                          for the process (V36)
GetProgramDir()()                       - Returns a lock on the directory
                                          the program was loaded
GetProgramName(buf,len)(d1/d2)          - Returns the current program name
                                          (V36)
GetPrompt(buf,len)(d1/d2)               - Returns the prompt for the
                                          current process (V36)
GetVar(name,buffer,size,flags)(d1/d2/d3/d4)
                                        - Returns the value of a local or
                                          global variable (V36)
Inhibit(name,onoff)(d1/d2)              - Inhibits access to a filesystem
                                          (V36)
InternalLoadSeg(fh,table,funcarray,stack)(d0/a0/a1/a2)
                                        - Low-level load routine (V36)
InternalUnLoadSeg(seglist,freefunc)(d1/a1)
                                        - Unloads a seglist loaded with
                                          InternalLoadSeg() (V36)
IsFileSystem(name)(d1)                  - Returns whether a Dos handler is
                                          a filesystem (V36)
LockDosList(flags)(d1)                  - Locks the specified Dos Lists
                                          for use (V36)
LockRecord(fh,offset,length,mode,timeout)(d1/d2/d3/d4/d5)
                                        - Locks a portion of a file (V36)
LockRecords(recArray,timeout)(d1/d2)    - Lock a series of records (V36)
MakeDosEntry(name,type)(d1/d2)          - Creates a DosList structure (V36)

MakeLink(name,dest,soft)(d1/d2/d3)      - Creates a filesystem link (V36)
MatchEnd(anchor)(d1)                    - Free storage allocated for
                                          MatchFirst()/MatchNext() (V36)
MatchFirst(pat,anchor)(d1/d2)           - Finds file that matches pattern
                                          (V36)
MatchNext(anchor)(d1)                   - Finds the next file or directory
                                          that matches pattern (V36)
MatchPattern(pat,str)(d1/d2)            - Checks for a pattern match with
                                          a string (V36)
MatchPatternNoCase(pat,str)(d1/d2)      - Checks for a pattern match with
                                          a string (V37)
MaxCli()()                              - Returns the highest CLI process
                                          number possibly in use (V36)
NameFromFH(fh,buffer,len)(d1/d2/d3)     - Get the name of an open
                                          filehandle (V36)
NameFromLock(lock,buffer,len)(d1/d2/d3) - Returns the name of a locked
                                          object (V36)
NewLoadSeg(file,tags)(d1/d2)            - Improved version of LoadSeg for
                                          stacksizes (V36)
NextDosEntry(dlist,flags)(d1/d2)        - Get the next Dos List entry (V36)

OpenFromLock(lock)(d1)                  - Opens a file you have a lock on
                                          (V36)
ParentOfFH(fh)(d1)                      - Returns a lock on the parent
                                          directory of a file (V36)
ParsePattern(pat,buf,buflen)(d1/d2/d3)  - Create a tokenized string for
```

```
                                          MatchPattern() (V36)
ParsePatternNoCase(pat,buf,buflen)(d1/d2/d3)
                                        - Create a tokenized string for
PathPart(path)(d1)                      - Returns a pointer to the end of
                                          the next-to-last (V36)
PrintFault(code,header)(d1/d2)          - Returns the text associated with
                                          a DOS error code (V36)
PutStr(str)(d1)                         - Writes a string the the default
                                          output (buffered) (V36)
ReadArgs(template,array,args)(d1/d2/d3) - Parse the command line input
                                          (V36)
ReadItem(name,maxchars,cSource)(d1/d2/d3)
                                        - Reads a single argument/name
                                          from command line (V36)
ReadLink(port,lock,path,buffer,size)(d1/d2/d3/d4/d5)
                                        - Reads the path for a soft
                                          filesystem link (V36)
Relabel(drive,newname)(d1/d2)           - Change the volume name of a
                                          volume (V36)
RemAssignList(name,lock)(d1/d2)         - Remove an entry from a multi-dir
                                          assign (V36)
RemDosEntry(dlist)(d1)                  - Removes a Dos List entry from
                                          it's list (V36)
RemSegment(seg)(d1)                     - Removes a resident segment from
                                          the resident list (V36)
ReplyPkt(dp,res1,res2)(d1/d2/d3)        - Replies a packet to the person
                                          who sent it to you (V36)
RunCommand(seg,stack,paramptr,paramlen)(d1/d2/d3/d4)
                                        - Runs a program using the current
                                          process (V36)
SameDevice(lock1,lock2)(d1/d2)          - Are two locks are on partitions
                                          of the device? (V37)
SameLock(lock1,lock2)(d1/d2)            - Returns whether two locks are on
                                          the same object (V36)
SelectInput(fh)(d1)                     - Select a filehandle as the
                                          default input channel (V36)
SelectOutput(fh)(d1)                    - Select a filehandle as the
                                          default input channel (V36)
SendPkt(dp,port,replyport)(d1/d2/d3)    - Sends a packet to a handler (V36)

SetArgStr(string)(d1)                   - Sets the arguments for the
                                          current process (V36)
SetConsoleTask(task)(d1)                - Sets the default console for the
                                          process (V36)
SetCurrentDirName(name)(d1)             - Sets the directory name for the
                                          process (V36)
SetFileDate(name,date)(d1/d2)           - Sets the modification date for a
                                          file or dir (V36)
SetFileSize(fh,pos,mode)(d1/d2/d3)      - Sets the size of a file (V36)
SetFileSysTask(task)(d1)                - Sets the default filesystem for
                                          the process (V36)
SetIoErr(result)(d1)                    - Sets the value returned by
                                          IoErr() (V36)
SetMode(fh,mode)(d1/d2)                 - Set the current behavior of a
                                          handler (V36)
SetProgramDir(lock)(d1)                 - Sets the directory returned by
                                          GetProgramDir (V36)
```

```
SetProgramName(name)(d1)                - Sets the name of the program
                                          being run (V36)
SetPrompt(name)(d1)                     - Sets the CLI/shell prompt for
                                          the current process (V36)
SetVar(name,buffer,size,flags)(d1/d2/d3/d4)
                                        - Sets a local or environment
                                          variable (V36)
SetVBuf(fh,buff,type,size)(d1/d2/d3/d4) - Set buffering modes and size
                                          (V36)
SplitName(name,seperator,buf,oldpos,size)(d1/d2/d3/d4/d5)
                                        - Splits out a component of a
                                          pathname into a buffer (V36)
StartNotify(notify)(d1)                 - Starts notification on a file or
                                          directory (V36)
StrToDate(datetime)(d1)                 - Converts a string to a DateStamp
                                          (V36)
StrToLong(string,value)(d1/d2)          - String to long value (decimal)
                                          (V36)
SystemTagList(command,tags)(d1/d2)      - Have a shell execute a command
                                          line (V36)
UnGetC(fh,character)(d1/d2)             - Makes a char available for
                                          reading again. (buffered) (V36)
UnLockDosList(flags)(d1)                - Unlocks the Dos List (V36)
UnLockRecord(fh,offset,length)(d1/d2/d3)- Unlock a record (V36)
UnLockRecords(recArray)(d1)             - Unlock a list of records (V36)
VFPrintf(fh,format,argarray)(d1/d2/d3)  - Format and print a string to a
                                          file (buffered) (V36)
VFWritef(fh,format,argarray)(d1/d2/d3)  - Write a BCPL formatted string to
                                          a file (buffered) (V36)
VPrintf(format,argarray)(d1/d2)         - Format and print string
                                          (buffered) (V36)
WaitPkt()()                             - Waits for a packet to arrive at
                                          your pr_MsgPort (V36)
WriteChars(buf,buflen)(d1/d2)           - Writes bytes to the the default
                                          output (buffered) (V36)
```

1.10 exec.library (basename: _SysBase)

```
AllocVec(byteSize,requirements)(d0/d1)  - Allocate memory and keep track
                                          of the size  (V36)
CacheClearE(address,length,caches)(a0,d0/d1)
                                        - Cache clearing with extended
                                          control (V37)
CacheClearU()()                         - User callable simple cache
                                          clearing (V37)
CacheControl(cacheBits,cacheMask)(d0/d1)- Instruction & data cache control
CachePostDMA(address,length,flags)(a0/a1,d1)
                                        - Take actions after to hardware
                                          DMA  (V37)
CachePreDMA(address,length,flags)(a0/a1,d1)
                                        - Take actions prior to hardware
                                          DMA  (V37)
ColdReboot()()                          - Reboot the Amiga (V36)
CreateIORequest(port,size)(a0,d0)       - Create an IORequest structure
                                          (V36)
```

```
CreateMsgPort()()                          - Allocate and initialize a new
                                             message port  (V36)
DeleteIORequest(iorequest)(a0)             - Free a request made by
                                             CreateIORequest()  (V36)
DeleteMsgPort(port)(a0)                    - Free a message port created by
                                             CreateMsgPort  (V36)
FreeVec(memoryBlock)(a1)                   - Return AllocVec() memory to the
                                             system  (V36)
ObtainSemaphoreShared(sigSem)(a0)          - Gain shared access to a
                                             semaphore (V36)
StackSwap(newSize,newSP,newStack)(d0/d1/a0)
                                           - Exec supported method of
                                             replacing a task's stack.
```

1.11 expansion.library (basename: _ExpansionBase)

```
AddBootNode(bootPri,flags,deviceNode,configDev)(d0/d1/a0/a1)
                                           - Add a BOOTNODE to the system
                                             (V36)
```

1.12 gadtools.library (basename: _GadToolsBase) V36

```
CreateContext(glistptr)(a0)                - Create a place for GadTools
                                             context data. (V36)
CreateGadgetA(kind,gad,ng,taglist)(d0/a0/a1/a2)
                                           - Allocate and initialize a
                                             gadtools gadget. (V36)
CreateMenusA(newmenu,taglist)(a0/a1)       - Allocate and fill out a menu
                                             structure. (V36)
DrawBevelBoxA(rport,left,top,width,height,taglist)(a0,d0/d1/d2/d3/a1)
                                           - Draws a bevelled box. (V36)
FreeGadgets(gad)(a0)                       - Free a linked list of gadgets.
                                             (V36)
FreeMenus(menu)(a0)                        - Frees memory allocated by
                                             CreateMenusA(). (V36)
FreeVisualInfo(vi)(a0)                     - Return any resources taken by
                                             GetVisualInfo. (V36)
GetVisualInfoA(screen,taglist)(a0/a1)      - Get information GadTools needs
                                             for visuals. (V36)
GT_BeginRefresh(win)(a0)                   - Begin refreshing friendly to
                                             GadTools. (V36)
GT_EndRefresh(win,complete)(a0,d0)         - End refreshing friendly to
                                             GadTools. (V36)
GT_FilterIMsg(imsg)(a1)                    - Filter an IntuiMessage through
                                             GadTools. (V36)
GT_GetIMsg(iport)(a0)                      - Get an IntuiMessage, with
                                             GadTools processing. (V36)
GT_PostFilterIMsg(imsg)(a1)                - Return the unfiltered message
                                             after
GT_RefreshWindow(win,req)(a0/a1)           - Refresh all the GadTools
                                             gadgets. (V36)
GT_ReplyIMsg(imsg)(a1)                     - Reply a message obtained with
```

```
                                          GT_GetIMsg(). (V36)
GT_SetGadgetAttrsA(gad,win,req,taglist)(a0/a1/a2/a3)
                                        - Change the attributes of a
                                          GadTools gadget. (V36)
LayoutMenuItemsA(firstitem,vi,taglist)(a0/a1/a2)
                                        - Position all the menu items.
                                          (V36)
LayoutMenusA(firstmenu,vi,taglist)(a0/a1/a2)
                                        - Position all the menus and menu
                                          items. (V36)
```

1.13 graphics.library (basename: _GfxBase)

```
BitMapScale(bitScaleArgs)(a0)           - Perform raster scaling on a bit
                                          map. (V36)
CloseMonitor(monitorSpec)(a0)           - Close a MonitorSpec (V36)
EraseRect(rp,xMin,yMin,xMax,yMax)(a1,d0/d1/d2/d3)
                                        - Fill a defined rectangular area
                                          using the current BackFill hook.
                                          (V36)
ExtendFont(font,fontTags)(a0/a1)        - Ensure tf_Extension has been
                                          built for a font (V36)
FindDisplayInfo(displayID)(d0)          - Search for a record identified
                                          by a specific key (V36)
FontExtent(font,fontExtent)(a0/a1)      - Get the font attributes of the
                                          current font (V36)
GetDisplayInfoData(handle,buf,size,tagID,displayID)(a0/a1,d0/d1/d2)
                                        - Query DisplayInfo Record
                                          parameters (V36)
GetVPModeID(vp)(a0)                     - Get the 32 bit DisplayID from a
                                          ViewPort. (V36)
GfxAssociate(associateNode,gfxNodePtr)(a0/a1)
                                        - Associate a graphics extended
                                          node with a given pointer
GfxFree(gfxNodePtr)(a0)                 - Free a graphics extended data
                                          structure (V36)
GfxLookUp(associateNode)(a0)            - Find a graphics extended node
                                          associated with a given pointer
                                          (V36)
GfxNew(gfxNodeType)(d0)                 - Allocate a graphics extended
                                          data structure (V36)
ModeNotAvailable(modeID)(d0)            - Check to see if a DisplayID
                                          isn't available. (V36)
NextDisplayInfo(displayID)(d0)          - Iterate current displayinfo
                                          identifiers (V36)
OpenMonitor(monitorName,displayID)(a1,d0)
                                        - Open a named MonitorSpec (V36)
ReadPixelArray8(rp,xstart,ystart,xstop,ystop,array,temprp)
                                         (a0,d0/d1/d2/d3/a2,a1)
                                        - Read the pen number value of a
                                          rectangular array
ReadPixelLine8(rp,xstart,ystart,width,array,tempRP)(a0,d0/d1/d2/a2,a1)
                                        - Read the pen number value of a
                                          horizontal line
ScalerDiv(factor,numerator,denominator)(d0/d1/d2)
```

```
                                        - Get the scaling result that
                                          BitMapScale would. (V36)
StripFont(font)(a0)                     - Remove the tf_Extension from a
                                          font (V36)
TextExtent(rp,string,count,textExtent)(a1,a0,d0/a2)
                                        - Determine raster extent of text
                                          data. (V36)
TextFit(rp,string,strLen,textExtent,constrainingExtent,strDirection,
       constrainingBitWidth,constrainingBitHeight)(a1,a0,d0/a2/a3,d1/d2/d3)
                                        - Count characters that will fit
                                          in a given extent (V36)
VideoControl(colorMap,tagarray)(a0/a1)  - Modify the operation of a
                                          ViewPort's ColorMap (V36)
WeighTAMatch(reqTextAttr,targetTextAttr,targetTags)(a0/a1/a2)
                                        - Get a measure of how well two
                                          fonts match. (V36)
WritePixelArray8(rp,xstart,ystart,xstop,ystop,array,temprp)
                (a0,d0/d1/d2/d3/a2,a1)
                                        - Write the pen number value of a
                                          rectangular array
WritePixelLine8(rp,xstart,ystart,width,array,tempRP)(a0,d0/d1/d2/a2,a1)
                                        - Write the pen number value of a
                                          horizontal line
```

1.14 icon.library (basename: _IconBase)

```
DeleteDiskObject(name)(a0)              - Delete a Workbench disk object
                                          from disk.
GetDefDiskObject(type)(d0)              - Read default wb disk object from
                                          disk.       (V36)
GetDiskObjectNew(name)(a0)              - Read in a Workbench disk object
                                          from disk.
PutDefDiskObject(diskObject)(a0)        - Write disk object as the default
                                          for its type.  (V36)
```

1.15 iffparse.library (basename: _IFFParseBase) V36

```
AllocIFF()()                            - Create a new IFFHandle
                                          structure.
AllocLocalItem(type,id,ident,dataSize)(d0/d1/d2/d3)
                                        - Create a local context item
                                          structure.
CloseClipboard(clipboard)(a0)           - Close and free an open
                                          ClipboardHandle.
CloseIFF(iff)(a0)                       - Close an IFF context.
CollectionChunk(iff,type,id)(a0,d0/d1)  - Declare a chunk type for
                                          collection.
CollectionChunks(iff,propArray,nProps)(a0/a1,d0)
                                        - Declare many collection chunks
                                          at once.
CurrentChunk(iff)(a0)                   - Get context node for current
                                          chunk.
```

```
EntryHandler(iff,type,id,position,handler,object)(a0,d0/d1/d2/a1/a2)
                                        - Add an entry handler to the
                                          IFFHandle context.
ExitHandler(iff,type,id,position,handler,object)(a0,d0/d1/d2/a1/a2)
                                        - Add an exit handler to the
                                          IFFHandle context.
FindCollection(iff,type,id)(a0,d0/d1)   - Get a pointer to the current
                                          list of collection
FindLocalItem(iff,type,id,ident)(a0,d0/d1/d2)
                                        - Return a local context item from
                                          the context stack.
FindProp(iff,type,id)(a0,d0/d1)         - Search for a stored property
                                          chunk.
FindPropContext(iff)(a0)                - Get the property context for the
                                          current state.
FreeIFF(iff)(a0)                        - Deallocate an IFFHandle struct.
FreeLocalItem(localItem)(a0)            - Deallocate a local context item
                                          structure.
GoodID(id)(d0)                          - Test if an identifier follows
                                          the IFF 85 specification.
GoodType(type)(d0)                      - Test if a type follows the IFF
                                          85 specification.
IDtoStr(id,buf)(d0/a0)                  - Convert a longword identifier to
                                          a null-terminated string.
InitIFF(iff,flags,streamHook)(a0,d0/a1) - Initialize an IFFHandle struct
                                          as a user stream.
InitIFFasClip(iff)(a0)                  - Initialize an IFFHandle as a
                                          clipboard stream.
InitIFFasDOS(iff)(a0)                   - Initialize an IFFHandle as a DOS
                                          stream.
LocalItemData(localItem)(a0)            - Get pointer to user data for
                                          local context item.
OpenClipboard(unitNum)(d0)              - Create a handle on a clipboard
                                          unit.
OpenIFF(iff,rwMode)(a0,d0)              - Prepare an IFFHandle to read or
                                          write a new IFF stream.
ParentChunk(contextNode)(a0)            - Get the nesting context node for
                                          the given chunk.
ParseIFF(iff,control)(a0,d0)            - Parse an IFF file from an
                                          IFFHandle struct stream.
PopChunk(iff)(a0)                       - Pop top context node off context
                                          stack.
PropChunk(iff,type,id)(a0,d0/d1)        - Specify a property chunk to
                                          store.
PropChunks(iff,propArray,nProps)(a0/a1,d0)
                                        - Declare many property chunks at
                                          once.
PushChunk(iff,type,id,size)(a0,d0/d1/d2)- Push a new context node on the
                                          context stack.
ReadChunkBytes(iff,buf,size)(a0/a1,d0)  - Read bytes from the current
                                          chunk into a buffer.
ReadChunkRecords(iff,buf,bytesPerRecord,nRecords)(a0/a1,d0/d1)
                                        - Read record elements from the
                                          current chunk into
SetLocalItemPurge(localItem,purgeHook)(a0/a1)
                                        - Set purge vector for a local
                                          context item.
```

```
StopChunk(iff,type,id)(a0,d0/d1)        - Declare a chunk which should
                                          cause ParseIFF to return.
StopChunks(iff,propArray,nProps)(a0/a1,d0)
                                        - Declare many stop chunks at
                                          once.
StopOnExit(iff,type,id)(a0,d0/d1)       - Declare a stop condition for
                                          exiting a chunk.
StoreItemInContext(iff,localItem,contextNode)(a0/a1/a2)
                                        - Store local context item in
                                          given context node.
StoreLocalItem(iff,localItem,position)(a0/a1,d0)
                                        - Insert a local context item into
                                          the context stack.
WriteChunkBytes(iff,buf,size)(a0/a1,d0) - Write data from a buffer into
                                          the current chunk.
WriteChunkRecords(iff,buf,bytesPerRecord,nRecords)(a0/a1,d0/d1)
                                        - Write records from a buffer to
                                          the current
```

1.16 input.device (basename: _InputBase)

```
PeekQualifier()()                       - Get the input device's current
                                          qualifiers (V36)
```

1.17 intuition.library (basename: _IntuitionBase)

```
AddClass(class)(a0)                     - Make a public class available
                                          (V36)
BuildEasyRequestArgs(window,easyStruct,idcmp,args)(a0/a1,d0/a3)
                                        - Simple creation of system
                                          request. (V36)
ChangeWindowBox(window,left,top,width,height)(a0,d0/d1/d2/d3)
                                        - Change window position and
                                          dimensions. (V36)
DisposeObject(object)(a0)               - Deletes a 'boopsi' object. (V36)
DrawImageState(rp,image,leftOffset,topOffset,state,drawInfo)
                                         (a0/a1,d0/d1/d2/a2)
                                        - Draw an (extended) Intuition
                                          Image with
EasyRequestArgs(window,easyStruct,idcmpPtr,args)(a0/a1/a2/a3)
                                        - Easy alternative to AutoRequest()
                                         . (V36)
EraseImage(rp,image,leftOffset,topOffset)(a0/a1,d0/d1)
                                        - Erases an Image. (V36)
FreeClass(classPtr)(a0)                 - Frees a boopsi class created by
                                          MakeClass(). (V36)
FreeScreenDrawInfo(screen,drawInfo)(a0/a1)
                                        - Finish using a DrawInfo
                                          structure. (V36)
GadgetMouse(gadget,gInfo,mousePoint)(a0/a1/a2)
                                        - Calculate gadget-relative mouse
                                          position. (V36)
```

```
GetAttr(attrID,object,storagePtr)(d0/a0/a1)
                                        - Inquire the value of some
                                          attribute of an object. (V36)
GetDefaultPubScreen(nameBuffer)(a0)     - Get name of default public
                                          screen. (V36)
GetScreenDrawInfo(screen)(a0)           - Get pointer to rendering
                                          information. (V36)
LockPubScreen(name)(a0)                 - Prevent a public screen from
                                          closing. (V36)
LockPubScreenList()()                   - Prevent changes to the system
                                          list. (V36)
MakeClass(classID,superClassID,superClassPtr,instanceSize,flags)
                                         (a0/a1/a2,d0/d1)
                                        - Create and initialize a boopsi
                                          class. (V36)
MoveWindowInFrontOf(window,behindWindow)(a0/a1)
                                        - Arrange the relative depth of a
                                          window. (V36)
NewObjectA(class,classID,tagList)(a0/a1/a2)
                                        - Create an object from a class.
                                          (V36)
NextObject(objectPtrPtr)(a0)            - Iterate through the object on an
                                          Exec list. (V36)
NextPubScreen(screen,namebuf)(a0/a1)    - Identify next public screen in
                                          the cycle. (V36)
ObtainGIRPort(gInfo)(a0)                - Set up a RastPort for a custom
                                          gadget. (V36)
OpenScreenTagList(newScreen,tagList)(a0/a1)
                                        - Also stack-based amiga.lib stub
                                          OpenScreenTags().  OpenScreen() with  ↩
                                             TagItem extension array. (V36)
OpenWindowTagList(newWindow,tagList)(a0/a1)
                                        - Also stack-based amiga.lib stub
                                          OpenWindowTags().  OpenWindow() with  ↩
                                             TagItem extension. (V36)
PointInImage(point,image)(d0/a0)        - Tests whether an image
                                          "contains" a point. (V36)
PubScreenStatus(screen,statusFlags)(a0,d0)
                                        - Change status flags for a public
                                          screen. (V36)
QueryOverscan(displayID,rect,oScanType)(a0/a1,d0)
                                        - Inquire about a standard
                                          overscan region. (V36)
ReleaseGIRPort(rp)(a0)                  - Release a custom gadget
                                          RastPort. (V36)
RemoveClass(classPtr)(a0)               - Make a public boopsi class
                                          unavailable. (V36)
ResetMenuStrip(window,menu)(a0/a1)      - Re-attach a menu strip to a
                                          window. (V36)
SetAttrsA(object,tagList)(a0/a1)        - Specify attribute values for an
                                          object. (V36)
SetDefaultPubScreen(name)(a0)           - Choose a new default public
                                          screen. (V36)
SetEditHook(hook)(a0)                   - Set global processing for string
                                          gadgets. (V36)
SetGadgetAttrsA(gadget,window,requester,tagList)(a0/a1/a2/a3)
                                        - Specify attribute values for a
```

```
                                          boopsi gadget. (V36)
SetMouseQueue(window,queueLength)(a0,d0)- Change limit on pending mouse
                                          messages. (V36)
SetPubScreenModes(modes)(d0)            - Establish global public screen
                                          behavior. (V36)
SysReqHandler(window,idcmpPtr,waitInput)(a0/a1,d0)
                                        - Handle system requester input.
                                          (V36)
UnlockPubScreen(name,screen)(a0/a1)     - Release lock on a public screen.
                                          (V36)
UnlockPubScreenList()()                 - Release public screen list
                                          semaphore. (V36)
ZipWindow(window)(a0)                   - Change window to "alternate"
                                          position and
```

1.18 keymap.library (basename: _KeymapBase)

```
AskKeyMapDefault()()                    - Ask for a pointer to the current
                                          default
MapANSI(string,count,buffer,length,keyMap)(a0,d0/a1,d1/a2)
                                        - Encode an ANSI string into
                                          keycodes. (V36)
MapRawKey(event,buffer,length,keyMap)(a0/a1,d1/a2)
                                        - Decode single raw key input
                                          event to an ANSI
SetKeyMapDefault(keyMap)(a0)            - Set the current default keymap.
                                          (V36)
```

1.19 layers.library (basename: _LayersBase)

```
CreateBehindHookLayer(li,bm,x0,y0,x1,y1,flags,hook,bm2)
                                         (a0/a1,d0/d1/d2/d3/d4/a3,a2)
                                        - Create a new layer behind all
                                          existing layers.
CreateUpfrontHookLayer(li,bm,x0,y0,x1,y1,flags,hook,bm2)
                                         (a0/a1,d0/d1/d2/d3/d4/a3,a2)
                                        - Create a new layer on top of
                                          existing layers.
InstallLayerHook(layer,hook)(a0/a1)     - Safely install a new
                                          Layer->BackFill hook.
MoveSizeLayer(layer,dx,dy,dw,dh)(a0,d0/d1/d2/d3)
                                        - Position/Size layer
```

1.20 mathieeesingbas.library (basename: _MathIeeeSingBasBase) V36

```
IEEESPAbs(parm)(d0)                     - Compute absolute value of IEEE
                                          single precision argument
IEEESPAdd(leftParm,rightParm)(d0/d1)    - Add one single precision IEEE
                                          number to another
IEEESPCeil(parm)(d0)                    - Compute Ceil function of IEEE
```

```
                                        single precision number
IEEESPCmp(leftParm,rightParm)(d0/d1)    - Compare two single precision
                                        floating point numbers
IEEESPDiv(dividend,divisor)(d0/d1)      - Divide one single precision IEEE
                                        by another
IEEESPFix(parm)(d0)                     - Convert IEEE single float to
                                        integer
IEEESPFloor(parm)(d0)                   - Compute Floor function of IEEE
                                        single precision number
IEEESPFlt(integer)(d0)                  - Convert integer to IEEE single
                                        precision number
IEEESPMul(leftParm,rightParm)(d0/d1)    - Multiply one double precision
                                        IEEE number by another
IEEESPNeg(parm)(d0)                     - Compute negative value of IEEE
                                        single precision number
IEEESPSub(leftParm,rightParm)(d0/d1)    - Subtract one single precision
                                        IEEE number from another
IEEESPTst(parm)(d0)                     - Compare IEEE single precision
                                        value to 0.0
```

1.21 mathieeesingtrans.library (basename: _MathIeeeSingTransBase) V36

```
IEEESPAcos(parm)(d0)                    - Compute the arc cosine of a
                                        number
IEEESPAsin(parm)(d0)                    - Compute the arcsine of a number
IEEESPAtan(parm)(d0)                    - Compute the arc tangent of
                                        number
IEEESPCos(parm)(d0)                     - Compute the cosine of a floating
                                        point number
IEEESPCosh(parm)(d0)                    - Compute the hyperbolic cosine of
                                        a floating point number
IEEESPExp(parm)(d0)                     - Compute the exponential of e
IEEESPFieee(parm)(d0)                   - Convert IEEE single to IEEE
                                        single
IEEESPLog(parm)(d0)                     - Compute the natural logarithm of
                                        a floating point number
IEEESPLog10(parm)(d0)                   - Compute logarithm base 10 of a
                                        number
IEEESPPow(exp,arg)(d1,d0)               - Raise a number to another number
                                        power
IEEESPSin(parm)(d0)                     - Compute the sine of a floating
                                        point number
IEEESPSincos(cosptr,parm)(a0,d0)        - Compute the arc tangent of a
                                        floating point number
IEEESPSinh(parm)(d0)                    - Compute the hyperbolic sine of a
                                        floating point number
IEEESPSqrt(parm)(d0)                    - Compute the square root of a
                                        number
IEEESPTan(parm)(d0)                     - Compute the tangent of a
                                        floating point number
IEEESPTanh(parm)(d0)                    - Compute the hyperbolic tangent
                                        of a floating point number
IEEESPTieee(parm)(d0)                   - Convert IEEE single to IEEE
                                        single
```

1.22 ramdrive.device (basename: _RamdriveDevice)

```
KillRAD(unit)(d0)                       - Kill ramdrive.device unit
KillRAD0()()                            - Kill ramdrive.device unit 0
```

1.23 rexxsyslib.library (basename: _RexxSysBase) V36

```
ClearRexxMsg(msgptr,count)(a0,d0)       - Releases and clears the argument
                                          array in a RexxMsg
CreateArgstring(string,length)(a0,d0)   - Create an argument string
                                          structure
CreateRexxMsg(port,extension,host)(a0/a1,d0)
                                        - Create an ARexx message
                                          structure
DeleteArgstring(argstring)(a0)          - Releases an Argstring created by
                                          CreateArgstring()
DeleteRexxMsg(packet)(a0)               - Releases a RexxMsg structure
                                          created by CreateRexxMsg()
FillRexxMsg(msgptr,count,mask)(a0,d0/d1)- Fill the argument strings as
                                          needed
IsRexxMsg(msgptr)(a0)                   - Function to determine if a
                                          message came from ARexx
LengthArgstring(argstring)(a0)          - Returns the length value stored
                                          in the argstring
LockRexxBase(resource)(d0)              - Obtain a semaphore lock on the
                                          RexxBase structure
UnlockRexxBase(resource)(d0)            - Release a semaphore lock on the
                                          RexxBase structure
```

1.24 Timer.Device (basename: _TimerBase)

```
GetSysTime(dest)(a0)                    - Get the system time. (V36)
ReadEClock(dest)(a0)                    - Get the current value of the
                                          E-Clock. (V36)
```

1.25 trackdisk.device (device commands)

```
TD_GETGEOMETRY                          - Gets the disk geometry table.
TD_EJECT                                - For those drives that support
                                          it.
```

1.26 utility.library (basename: _UtilityBase) V36

```
AllocateTagItems(numItems)(d0)          - Allocate a TagItem array (or
                                          chain). (V36)
Amiga2Date(amigaTime,date)(d0/a0)       - Calculate the date from a
                                          timestamp.  (V36)
```

```
CallHookPkt(hook,object,paramPacket)(a0/a2,a1)
                                        - Invoke a Hook function callback.
                                          (V36)
CheckDate(date)(a0)                     - Checks ClockData struct for
                                          legal date. (V36)
CloneTagItems(tagList)(a0)              - Copies a TagItem list.  (V36)
Date2Amiga(date)(a0)                    - Calculate seconds from
                                          01-Jan-1978.  (V36)
FilterTagChanges(newTagList,oldTagList,apply)(a0/a1,d0)
                                        - Eliminate TagItems which specify
                                          no change. (V36)
FilterTagItems(tagList,filterArray,logic)(a0/a1,d0)
                                        - Remove selected items from a
                                          TagItem list. (V36)
FindTagItem(tagVal,tagList)(d0/a0)      - Scans TagItem list for a Tag.
                                          (V36)
FreeTagItems(tagList)(a0)               - Frees allocated TagItem lists.
                                          (V36)
GetTagData(tagVal,defaultVal,tagList)(d0/d1/a0)
                                        - Obtain data corresponding to
                                          Tag. (V36)
MapTags(tagList,mapList,includeMiss)(a0/a1,d0)
                                        - Convert ti_Tag values in a list
                                          via map pairing. (V36)
NextTagItem(tagListPtr)(a0)             - Iterate TagItem lists. (V36)
PackBoolTags(initialFlags,tagList,boolMap)(d0/a0/a1)
                                        - Builds a "Flag" word from a
                                          TagList. (V36)
RefreshTagItemClones(cloneList,origList)(a0/a1)
                                        - Rejuvenates a clone from the
                                          original. (V36)
SDivMod32(dividend,divisor)(d0/d1)      - Signed 32 by 32 bit division and
                                          modulus. (V36)
SMult32(factor1,factor2)(d0/d1)         - Signed 32 by 32 bit multiply
                                          with 32 bit result. (V36)
Stricmp(string1,string2)(a0/a1)         - Case-insensitive string compare.
                                          (V37)
Strnicmp(string1,string2,length)(a0/a1,d0)
                                        - Case-insensitive string compare,
                                          length-limited. (V37)
TagInArray(tagVal,tagArray)(d0/a0)      - Check if a Tag value appears in
                                          a Tag array. (V36)
ToLower(character)(d0)                  - Convert a character to
                                          lowercase. (V37)
ToUpper(character)(d0)                  - Convert a character to
                                          uppercase. (V37)
UDivMod32(dividend,divisor)(d0/d1)      - Unsigned 32 by 32 bit division
                                          and modulus. (V36)
UMult32(factor1,factor2)(d0/d1)         - Unsigned 32 by 32 bit multiply
                                          with 32 bit result. (V36)
```

1.27 workbench.library (basename: _WorkbenchBase)

```
AddAppIconA(id,userdata,text,msgport,lock,diskobj,taglist)
                                         (d0/d1/a0/a1/a2/a3/a4)
```

```
                                        - Also stack-based amiga.lib stub
                                          AddAppIcon().  Add an icon to workbench' ↩
                                             s list of appicons.        (V36)
AddAppMenuItemA(id,userdata,text,msgport,taglist)(d0/d1/a0/a1/a2)
                                        - Also stack-based amiga.lib stub
                                          AddAppMenuItem().  Add a menuitem to  ↩
                                             workbench's list of appmenuitems (V36 ↩
                                             )
AddAppWindowA(id,userdata,window,msgport,taglist)(d0/d1/a0/a1/a2)
                                        - Also stack-based amiga.lib stub
                                          AddAppWindow() add a window to workbench ↩
                                             's list of appwindows.  (V36)
RemoveAppIcon(appIcon)(a0)              - Remove an icon from workbench's
                                          list            (V36)
RemoveAppMenuItem(appMenuItem)(a0)      - Remove a menuitem from
                                          workbench's list      (V36)
RemoveAppWindow(appWindow)(a0)          - Remove a window from workbench's
                                          list         (V36)
```