

So Solomon overlaid the house within with pure gold: and he made a partition by the chains of gold before the oracle; and he overlaid it with gold.

And the whole house he overlaid with gold, until he had finished all the house: also the whole altar that was by the oracle he overlaid with gold.

And within the oracle he made two cherubims of olive tree, each ten cubits high.

1 Kings 6:21 (King James)



# TempleOS

Jihong Lee  
13 March 2022

O/S with Duane

# Biography of Terry A. Davis: The Greatest Programmer to Ever Live

Articles in **Operating Systems and Low-Level Systems** | By **Hash Brown**

*Published 3 years ago / Last Update 3 years ago*

*The most comprehensive source and biography of information you will find on the TempleOS creator, Terry A. Davis.*



STEAM GROUP

## TempleOS Fan Club

344

MEMBERS

28

IN-GAME

142

ONLINE

Founded

April 1, 2018

Overview



## Quick Intro

```
Sat 06/15 08:25:01 Ref:63 Mem:0156A9EA00 CPU19_1 1 1 1 1 1 1 CTRL ALT T=7FD8662LXJ
MENU=#1 .....=7FD88C28=LXJ MENU=#2 Task...Unnamed T=7FD8662LXJ
```

TempleOS  
Trivial Solutions

-] Keyboard Ctrl's

Enter Right-click menu  
Space Bar Edit (view)  
Esc Save and exit  
Shift-Esc Abort  
Ctrl-m Personal Menu  
Windows Key P=MENU  
F1 HT  
Other keys: Key

-] Mouse Ctrl's

Right-Click  
Left-Click  
Double-Left  
Double-Right  
Top of Screen

-] Keyboard-Mouse

Ctrl-LeftDrag  
Ctrl-Right  
Ctrl-Alt-z  
Ctrl-Alt-Z

Remember to termi  
semicolon.

Help Index Quick

Directory of D:/T

DATE TIME SIZE  
04/26 02:03 0000  
04/26 02:03 0000  
04/26 02:03 0000  
05/10 23:21 0104  
05/10 23:21 0134  
06/15 20:49 01EC  
06/14 21:33 0066  
06/15 20:31 143F  
DFT: 8 Cores 3.39  
0.000000s ans0=0x  
D:/TempleOS/Accts  
tOnce";  
0.000000s ans0=0x  
D:/TempleOS/Accts

TempleOS  
Trivial Solutions

-] Keyboard Ctrl's

Enter Right-click menu  
Space Bar Edit (view)  
Esc Save and exit  
Shift-Esc Abort  
Ctrl-m Personal Menu  
Ctrl-m down Menu


-click menu  
(view)  
and exit  
  
down menu  
  
ls  
Scroll Window  
scrolling  
screen on Mouse  
m screen on Mouse  
  
cmds with a

rt--Command Line

eOS/Accts/TS/TAD

gOrgan  
3.CPP.Z  
3alocalize.CPP.Z  
stry.CPP.Z  
.CPP.Z  
onalNotes.TXT.Z  
  
0000=0  
TAD>■

TempleOS V1.03



Public Domain  
Operating System

Trivial Solutions

Tip of day  
\* You can use ans0,ans1, etc. in cmd line  
expressions. They hold results from the  
last cmd line operations. You can use the  
cmd prompt as a calculator by just  
entering expressions like 1+2\*3;. F64  
results can be accessed with ansf0 or ansf  
1.  
<SHIFT-ESC> to exit.

F0F

F0F

```

Mon 06/08 11:55:40 FPS:31 Mem:00160E9200 CPU 1 PRTSCRN2
MENU "C:/Kernel";Di=16F14A28
DATE TIME SIZE
06/07 10:39 000000 +-
06/07 10:39 000000 +-
06/07 12:50 0038E MakeDistro.CPP.Z
06/08 11:54 01F9 Registry.CPP.Z
0.000150 ans=0x00000004=4
C:/Home>Cd("C:/Kernel");Dir;
Directory of C:/Kernel
DATE TIME SIZE
06/07 10:39 000000 +-
06/07 10:39 000000 +-
06/07 10:39 000000 task
06/07 10:39 000000 KbdMouse
06/07 10:39 000000 Compress.CPP.Z
06/07 12:52 0002E Date.CPP.Z
06/07 12:52 00822 DateLib.CPP.Z
06/07 12:52 000CF dLite.CPP.Z
06/07 12:52 0084C Font8x8.CPP.Z
06/07 12:46 00AF8 Ints.CPP.Z
06/07 12:52 00F5F XCfg.CPP.Z
06/07 14:47 00A06 DataTypes.CPP.Z
06/07 14:47 01C98 Dbg.CPP.Z
06/07 07:50 00DC6 Define.CPP.Z
06/07 11:54 00E15 End.CPP.Z
06/07 12:46 007F4 Except.CPP.Z
06/07 14:02 007DC ExtIs.CPP.Z
06/07 14:47 00263 GlibIs.CPP.Z
06/07 12:46 00C53 Hash8.CPP.Z
06/07 14:47 00D90 Hash8.CPP.Z
06/07 14:47 00CA9 Load.CPP.Z
06/07 12:46 00BEF Math8.CPP.Z
06/07 14:47 00590 Math8.CPP.Z
06/07 12:52 00D32 Misc.CPP.Z
06/07 13:07 0148D Start.CPP.Z
06/07 12:44 008B5 Utils.CPP.Z
06/07 14:47 00ACH XevDev.CPP.Z
06/07 14:47 017E0 Xemb.CPP.Z
06/07 13:57 0130A Xemb.CPP.Z
06/07 14:47 00F53 Xemb.CPP.Z
06/07 12:52 01BDF Xlib.CPP.Z
06/07 12:52 0084A XlibIOS.CPP.Z
06/07 12:52 00DA6 Xsect.CPP.Z
06/07 12:52 00513 Xsort.CPP.Z
06/07 12:52 011D2 Xsect.CPP.Z
06/07 12:52 0176E XrvCmd.CPP.Z
06/07 12:43 0177D Xstr.CPP.Z
06/07 14:47 0088A Xstr.CPP.Z
06/07 14:47 01D04 XstrPrint.CPP.Z
06/07 14:47 00DA7 XstrScan.CPP.Z
06/07 12:52 01A05 Xtask.CPP.Z
06/07 14:26 0A96D Xkernel.CPP.Z
06/07 14:02 01969 Xkernel.CPP.Z
06/07 14:03 02D2D Xkernel.CPP.Z
06/07 10:39 14A6F Xkernel.CPP.Z
06/07 12:52 00380 Xkernel.CPP.Z
0.001600 ans=0x0000002E=46
C:/Kernel>

```

Uf("Foo")  
 disassembles function  
 Type()  
 displays files  
 Dir;  
 serves as file explorer

Sat 02/27 20:43:46 FPS:29 Mem:0015989A00 CPU24 1 1 1 1 1 1 1 ENTER

PMENU sonalMenu.D0.2.. [X] MENU sk...Unnamed T: 16563A28 [X]

Copy this file to your /Home directory and modify it. This version is the default.

cd Dir: L

cd (../Demo): Dir: L

cd (../Demo/Graphics): Dir: L

cd (../Demo/Snd): Dir: L

cd (../Demo/DolDoc): Dir: L

cd (../Demo/Games): Dir: L

cd (../Demo/MultiCore): Dir: L

cd (../Demo/Hsm): Dir: L

cd (../Doc): Dir: L

cd (../Demo/Lectures): Dir: L

cd (../Apps): Dir: L

Run IOSStdIns() if you wish to use the official staff /Home files.

cd (../Demo/AcctExample): Dir: L

Make all with BootHDIns().

cd (../Kernel): Dir: L

cd (../Compiler): Dir: L


The ../StartOS.HC file is compiled every time you boot.

cd (../Adam): Dir: L



Welcome Help & Index Demo Index

FastReboot Take Tour Key Map

**Fun Games**

Titanium Diamond Black Diamond Flap Bat Vaboom

FlatTops Bomber Golf KeepAway RawHide

More-TB Line:0030 Col:0021

Public Domain Operating System

Help & Index, Quick Start, Cmd line

Directory of D:/Home

DATE	TIME	SIZE
02/27	20:37	0000
02/27	20:37	0000
06/21	11:11	0000
01/10	23:01	0000
02/27	20:37	0000

Word: Tour

01 TourPut

02 Tour

03 TourMenu

04 TourPutTask

05 Tour5

06 Tours

07 Tour

08 Touraco

09 Tourbillion

10 Tourist

11 Tourmaline

12 Tourn

13 Tournament

14 Tournery

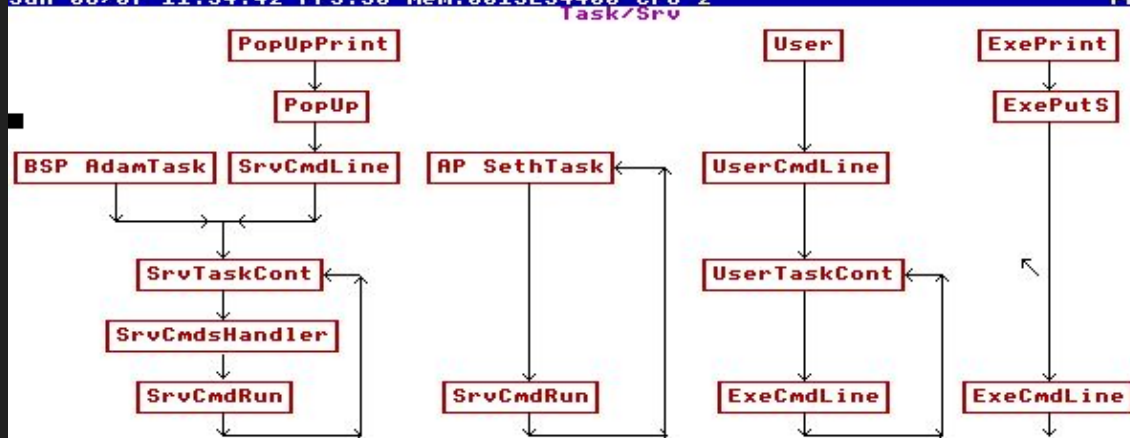
15 Tourney

16 Tourniquet

TX

EOF





CSrvCmd nodes are one of **five types**. User applications deal with text or msgs. SrvCmdRun() will call a function, spawn a task or execute some text src code.

```

#define SUCT_TEXT_INPUT      0 //TaskText()   Feed StdIn
#define SUCT_MSG             1 //TaskMsg()   Post msg
#define SUCT_EXE_STR         2 //TaskExe()   Compile & execute src code text
#define SUCT_CALL            3 //JobQue()    Tell MP to call function
#define SUCT_SPAWN_TASK      4 //Spawn()     Tell MP to spawn task
  
```

Several **other routines** include a call to **SrvCmdsHandler()** that gives them powerful ability to execute servant cmds.

	Task/Srv/Exe	
Adam	Funct	Sends text to the adam task which gets compiled and ru
AdamErr	Funct	Sends text to the adam task which gets displayed as re
AdamFile	Funct	Tell adam task to compile and run a file.
AdamLog	Funct	Sends text to the adam task which gets displayed.
ExeCmdLine	Funct	Terminal JIT-compile-and-execute loop for CCmpCtrl.
JobQue	Funct	Queue multicore jobs, handled by Seth tasks.
JobResultGet	Funct	See <b>./Demo/MultiCore/Lock.CPP</b>
JobResultScan	Funct	Check rqt complete, return with or without.
PopUp	Funct	Execute code text in PopUp win task.
PopUpEd	Funct	Create PopUp win task and edit a doc.
PopUpFile	Funct	Run code file in PopUp win task.
PopUpPrint	Funct	Execute code text in PopUp win task.
Spawn	Funct	Create task on core running at address.
SrvCmdDel	Funct	Free one cmd node.

```

Mon 06/08 12:00:28 FPS:31 Mem:00160A7C00 CPU 1
MENU /ClassMeta.CPP.2
/*Demonstrates class meta data.
Basically, we make use of the compiler's
data about a class. We can add to the
compilers data arbitrary items, either string or
or int or F64 (if you typecast).
This stuff is not high performance.
Don't get carried away -- it might be slow.
*/
U0 RankOut(I64 i)
{
    " %z", i, "Cadet\0Ensign\0Captain\0Admiral\0President\0";
}

class Test1Struct
{
    I64 age      print_str "%2d" dft_val 38;
    I64 color    dft_val RED; //Accepts expressions
    I64 rank     print_str "%1d" dft_val 6/2 output_fun &RankOut;
};

class Test2Struct
{
    I64 age      print_str "%2d" dft_val 38 percentile 54.20;
    I64 rank     print_str "%1d" dft_val 5;
    I64 serial_num print_str "%6d" dft_val 123456;
};

U0 DumpStruct(U8 *_d, U8 *class_name=lastclass)
    cdt=Now;
    "$IV,1$----%D %T----$IV,0$\n", cdt, cdt;

```

- lastclass keyword
- no ahead-of-time linker, object file
- only regular variables
- symbol table

- no main()
- all code JIT-compiled on demand
- function tagged with #help-index
- #exe compiler
- full metadata, reflection support

```

Bool MyPutKey(I64 ch, I64 sc)
{
    //ch=ASCII; sc=scan_code
    //See Char for definition of scan codes.
    //See Key Allocations.
    //See Keyboard Devices.

    //You can customize keys. This routine
    //is called before the main editor
    //key handler DocPutKey().
    //You can intercept any key.

    //Return TRUE if you completely
    //handled the key.

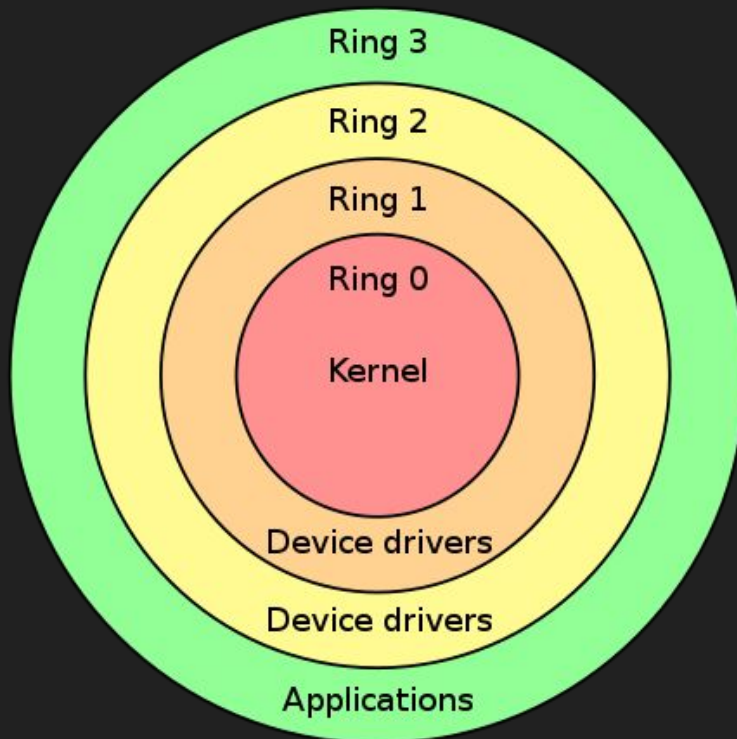
    I64 i, sc2=sc, u8[0];
    U8 *st1, *st2;

```



# Hardware & Security

- no hardware support
- no memory protection
- no file permissions
- no threads



**Why TempleOS?**

- <http://www.codersnotes.com/notes/a-constructive-look-at-templeos/>
- <https://hackmd.io/@dethe/H1kZI6IOU>
- <https://www.youtube.com/watch?v=LtlyeDAJR7A>
- <https://www.quora.com/Can-I-get-your-honest-opinion-on-Terry-A-Davis-and-TempleOS>
- <https://christine.website/blog/series/templeos>
- <https://apoorvupreti.com/templeos/>
- <https://templeos.org/>