

UEFI Basic Tutorial (Part 9) - Simple Use of EVENT

原创

xiaopangzi313

Posted on 2020-06-06 10:21:05

Read 3.8k

Collection 8

Likes 2

Copyright CC 4.0 BY-SA

Category columns: 15_Firmware Development

 15_Firmware Devel... This column includes this content

27 articles

Subscribe to our column

摘要

This article describes how to use the EVENT API to implement key and timer event monitoring in a UEFI environment. By creating and waiting for events, it demonstrates the application of asynchronous mechanisms in limited multitasking support.

The summary is generated in [C Know](#) , supported by DeepSeek-R1 full version, [go to experience](#)>

UEFI Basic Tutorial (VIII) - Simple Use of EVENT

1. Write source code

1. Write the UEFI Application
- code C:\edki\OvmfPkg\MyHelloWorldEvent\MyHelloWorldEvent.c,

C

AI generated projects

登录复制

run

```
1 EFI_STATUS MyHelloWorldEventEntry(  
2     IN EFI_HANDLE      ImageHandle,  
3     IN EFI_SYSTEM_TABLE *SystemTable  
4 )  
5 {  
6     EFI_STATUS Status;  
7  
8     UINTN      Index=0;  
9     EFI_INPUT_KEY Key;  
10    EFI_EVENT myEvents[2] = {0};  
11  
12  
13    Print (L"[MyHelloWorldEvent] MyHelloWorldEventEntry Start..\n");  
14    // 1. 生成事件  
15    // 生成按键事件  
16    myEvents[0] = gST->ConIn->WaitForKey;  
17  
18  
19  
20    // 生成Timer事件  
21    Status = gBS->CreateEvent(EVT_TIMER , TPL_CALLBACK, (EFI_EVENT_NOTIFY)NULL, (VOID*)NULL, &myEvents[1]);  
22    if(EFI_ERROR(Status)){  
23        Print (L"[MyHelloWorldEvent] CreateEvent %r ... \n",Status);  
24        return Status;  
25    }  
26  
27    Status = gBS->SetTimer(myEvents[1],TimerPeriodic , 100 * 1000 * 1000);//设置10秒定时  
28    if(EFI_ERROR(Status)){  
29        Print (L"[MyHelloWorldEvent] SetTimer %r ... \n",Status);  
30        return Status;  
31    }  
32  
33    while (1){  
34        //2. 阻塞并等待事件被触发  
35        Status = gBS->WaitForEvent(2, myEvents, &Index);
```

```

36     if(EFI_ERROR(Status)){
37         Print (L"[MyHelloWorldEvent] WaitForEvent %r ...\n",Status);
38         return Status;
39     }
40
41     if (Index == 0){
42         // 读取按键键值并显示
43         Status = gST->ConIn->ReadKeyStroke (gST->ConIn, &Key);
44         switch (Key.ScanCode){
45             case SCAN_UP:
46                 Print (L"[MyHelloWorldEvent] Key UP is Pressed..\n");
47                 break;
48
49             case SCAN_DOWN:
50                 Print (L"[MyHelloWorldEvent] Key Down is Pressed..\n");
51                 break;
52
53             ...
54
55             case SCAN_ESC:
56                 Print (L"[MyHelloWorldEvent] Key ESC is Pressed..\n");
57                 goto End;
58
59             default:
60                 Print (L"[MyHelloWorldEvent] Key %a is Pressed..\n", (CHAR8 *)&Key.UnicodeChar);
61                 break;
62         }
63     }else{
64         Print (L"[MyHelloWorldEvent] Timer event is trigered ..\n");
65     }
66 }
67
68 End:
69 //销毁事件
70 Status = gBS->CloseEvent(myEvents[0]);
71 Status = gBS->CloseEvent(myEvents[1]);
72
73 Print (L"[MyHelloWorldEvent] MyHelloWorldEventEntry End..\n");
74 return Status;
75 }

```

收起 ^

2. Compile and generate EFI files

Run and `edksetup.bat` compile the entire OvmfPkg Package

3. Run UEFI APP **MyHelloWorldEvent.efi**

1. Copy `C:\edkii\Build\OvmfX64\DEBUG_VS2013x86\FV\0VMF.fd` to `C:\qemu`; Copy `C:\edkii\Build\OvmfX64\DEBUG_VS2013x86\X64\OvmfPkg\MyHelloWorldEvent\MyHelloWorldEvent\OUTPUT\MyHelloWorldEvent.efi` to virtual disk `HDD_BOOT.img`
2. Execute `setup-qemu-x64.bat`, and then execute in `findstr MyHelloWorldEvent UEFI SHELL MyHelloWorldEvent.efi`

Machine View

```
PciRoot (0x0) /Pci (0x1,0x1) /Ata (0x0)
BLK0: Alias(s) :
PciRoot (0x0) /Pci (0x1,0x0) /Floppy (0x0)
BLK1: Alias(s) :
PciRoot (0x0) /Pci (0x1,0x0) /Floppy (0x1)
Press ESC in 1 seconds to skip startup.nsh or any other key to continue.
Shell> fs0:
FS0:\> MyHelloWorldEvent.efi
[MyHelloWorldEvent] MyHelloWorldEventEntry Start..
[MyHelloWorldEvent] Key UP is Pressed..
[MyHelloWorldEvent] Key Down is Pressed..
[MyHelloWorldEvent] Key LEFT is Pressed..
[MyHelloWorldEvent] Key RIGHT is Pressed..
[MyHelloWorldEvent] Key C is Pressed..
[MyHelloWorldEvent] Key S is Pressed..
[MyHelloWorldEvent] Key D is Pressed..
[MyHelloWorldEvent] Key N is Pressed..
[MyHelloWorldEvent] Key 2 is Pressed..
[MyHelloWorldEvent] Key 0 is Pressed..
[MyHelloWorldEvent] Key 1 is Pressed..
[MyHelloWorldEvent] Key 9 is Pressed..
[MyHelloWorldEvent] Key . is Pressed..
[MyHelloWorldEvent] Key . is Pressed..
[MyHelloWorldEvent] Timer event is triggered ..
```

<https://blog.csdn.net/xiaopangzi313>

IV. Summary

Events is another type of object managed by UEFI Service, which can provide an asynchronous mechanism to support limited multitasking, including keyboard, mouse, timer, protocol, etc. This article uses some APIs of Event to monitor key and timer events and make simple processing. Among them, CreateEvent is used to generate events, WaitForEvent is used to block and wait for events to be triggered, and CloseEvent is used to destroy events.

Event DEMO source code

用Scrapy给TikTok喂"流量兴奋剂"

广告

逆向工程竞品标签池：3天霸榜技术流 → 技术人优先预约通道

[about Us](#)[Careers](#)[Business Cooperation](#)[Seeking coverage](#)[400-660-0108](#)kefu@csdn.net[Online Customer Service](#)[Working hours 8:30-22:00](#)

Public Security Registration Number 11010502030143 Beijing ICP No. 19004658 Beijing Internet Publishing House [2020] No. 1039-165

Commercial website registration information Beijing Internet Illegal and Harmful Information Reporting Center Parental Control

Online 110 Alarm Service China Internet Reporting Center Chrome Store Download Account Management Specifications

Copyright and Disclaimer Copyright Complaints Publication License Business license

©1999-2025 Beijing Innovation Lezhi Network Technology Co., Ltd.