Windows 10 on ARM for Driver Developers

Divya Bhat
Agenda

The Always Connected PC
Evaluating Effort
Recompiling and Debugging
Signing and Packaging
Summary and Best Practices
Q&A
Always Connected PC

Windows 10

Amazing battery life

LTE Connectivity

Instant On
I traveled from Thursday – Sunday last week with only my ARM64 device. I used it for email, Twitter, web surfing, Skype meetings, a WebEx screen test, and even for looking at some code in Visual Studio. What I didn't do was recharge the battery. It resumes instantly when I want to use it, and with the SIM card it's always connected. – Ginny Caughey MVP
Always Connected PCs Available

- ASUS NovaGo TP370QL
- HP Envy X2
- Lenovo Miix 630
- Lenovo Yoga C630
- Snapdragon 835
- Snapdragon 850
- WinHEC 2018
It’s “Just Windows”

WinHEC 2018
User mode

- Shell + Cortana: Compiled for ARM64
- Edge: Compiled for ARM64
- Services: Compiled for ARM64
- 3rd Party Apps (non-UWP): X86
- 3rd Party Apps (UWP): X86, ARM32, ARM64

Kernel + Drivers

- Compiled for ARM64

- CPU
- Graphics
- Disk
- Networking
**x86 Emulation**

<table>
<thead>
<tr>
<th>Workload Type</th>
<th>Expected performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU-intensive</td>
<td>✓</td>
</tr>
<tr>
<td>I/O-intensive</td>
<td>✓ ✓</td>
</tr>
<tr>
<td>GPU-intensive</td>
<td>✓ ✓</td>
</tr>
</tbody>
</table>

**Native Process**
- Edge, Cortana, Shell...
- System Services (NTDLL)
- Native System DLLs

**Emulated Process**
- x86 app (e.g. Photoshop)
- WOW Abstraction Layer
- X86-to-ARM CPU Emulator
- System Services (NTDLL)
- CHPE DLLs

**Workload Type**

- **CPU-intensive**
- **I/O-intensive**
- **GPU-intensive**

**Expected performance**

- ✓
- ✓ ✓
- ✓ ✓

**Performance**
- CPU-intensive
- GPU-intensive
- I/O-intensive
Always Connected PC at a glance

Brings amazing battery life and connectivity
It’s still Windows (and runs x86 apps)
Drivers do need to be recompiled for ARM64
https://aka.ms/Win10onARM

Developer reference for Windows 10 on ARM
Evaluating Scope

**Driver-only projects**
Will need to be rebuilt as ARM64 using the latest tools
Possible to not need any code changes
Be aware of custom ASM code and 3rd party dependencies

**UI components**
Can remain x86 or move to ARM64
Highly dependent on workload and dependencies
Recompiling UI Components

Keep as x86 if:
Not particularly performance intensive or if I/O- or GPU-bound
Dependent on frameworks not yet available for ARM64

Rebuild as ARM64 if:
CPU Intensive
Need to match bitness of driver
Need to be loaded into a native process

WinHEC 2018
Visual Studio 2017

Update to the latest preview
Download the 15.9 preview at https://visualstudio.com/pre
Select C++ compilers and libraries for ARM/ARM64

Not already using VS2017?
Allows you to compile your code for ARM64
Produces faster and more secure code for other architectures too
Can still target down-level OS (e.g. Windows XP)
Can sign your drivers within VS 2017
Windows Driver Kit (WDK)
Version 1709 or later is required to support ARM64 drivers

Windows 10 SDK
Version 1709 or later is required to support ARM64 drivers
The most recent preview is recommended to support ARM64 applications
Demo

OpenVPN Recompilation
TAP-Windows -- A kernel driver to provide virtual tap device functionality on Windows.

This code was inspired by the CIPE-Win32 driver by Dawson K. Wilson.

This source code is Copyright (C) 2002-2014 OpenVPN Technologies, Inc., and is released under the GPL version 2 (see below).

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License version 2
as published by the Free Software Foundation.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software Foundation, Inc.,
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

#define __TAP_H
#define __TAP_H

Show output from: Driver
Driver output is listed here.
Get an ARM64 Device

Testing requires a physical ARM64 device
There are no ARM64 VMs or supported emulators today

Device availability
Always Connected PCs are on sale in many regions around the world
Reach out to us if you are blocked by device availability
Setup for Debugging

Kernel debugging support
KDNET is supported with USB NICs on all ARM64 devices
WinDbg supports ARM64
ARM64 symbols are available as with any other architecture

Visual Studio support
Remote debugging to an ARM64 target is supported in Visual Studio 2017

3rd party tools
More tools continue to add support for ARM64, such as DebugLogger
Attestation sign or WHQL sign

Test in ARM64 Mode, or ARM64 Mode + x86 on ARM Mode
HLK ARM64 Getting Started Guide

Setup and testing for ARM64 desktop machines largely follows the guidelines found in the HLK Getting Started Guide. Instructions specific to ARM64 are described in this topic.

This topic provides a basic introduction to using the Windows Hardware Lab Kit (HLK). Once you are familiar with the tools and process, you can review the HLK User's Guide for additional information.

Windows HLK Prerequisites

Prerequisites are the same for all desktop machines. See the HLK Prerequisites topic for more information.

Step 1: Install Controller and Studio on the test server
Attestation sign or WHQL sign

**HLK**

Test in ARM64 Mode, or ARM64 Mode + x86 on ARM Mode

*Not ARM-specific* W10 and EWDK test sign, but deploying requires MSFT sig

- **How** to sign across releases
Driver Signing changes in Windows 10, version 1607

Joshua Baxter    July 26, 2016

Last year, we announced that beginning with the release of Windows 10, all new Windows 10 kernel mode drivers must be submitted to the Windows Hardware Developer Center Dashboard portal (Dev Portal) to be digitally signed by Microsoft. However, due to technical and ecosystem readiness issues, this was not enforced by Windows Code Integrity and remained only a policy statement.

Starting with new installations of Windows 10, version 1607, the previously defined driver signing rules will be enforced by the Operating System, and Windows 10, version 1607 will not load any new kernel mode drivers which are not signed by the Dev Portal. OS signing enforcement is only for new OS installations; systems upgraded from an earlier OS to Windows 10, version 1607 will not be affected by this change.

We're making these changes to help make Windows more secure. These changes limit the risk of an end-user system being compromised by malicious driver software.

If you are a driver developer, here is what you need to do:

1. Ensure that you submit new drivers to Microsoft via the Windows Hardware Developer Center Dashboard portal.
2. Begin the process of getting an Extended Validation (EV) Code Signing Certificate. All drivers submitted to the portal must be signed by an EV certificate.

FAQs

What are the exact exceptions? Are cross-signed drivers still valid?

Enforcement only happens on fresh installations, with Secure Boot on, and only applies to new kernel mode drivers:
Shipping

Windows Update

ARM64 drivers are fully supported on Windows Update
Ensure a smooth experience for customers by submitting drivers when ready

Installers

If an app needs to install the correct driver, use the IsWow64Process2 API to detect ARM64
On a webpage, offer the correct download by checking window.navigator.userAgent
Summary and best practices

Always Connected PCs are here
Windows 10 on ARM enables amazing portable scenarios while still being Windows

Updating for ARM64 is doable
Updating drivers for ARM64 is worthwhile and may not take much effort
Make sure you’re on the latest version of the Visual Studio tools

Don’t forget
NDIS drivers need to support version 6.30 to run on ARM64
Use IsWow64Process2 to detect when running on ARM64
Christiano Aman, President of Qualcomm
Jerry Shen, CEO of ASUS

World’s 1st Gigabit LTE Laptop

Up to 1 Gbps
Qualcomm® Snapdragon™ X16 LTE modem

3-7x faster than average broadband speeds

* FCC Dec. 2016 - U.S. fixed broadband speed

© 2018 Microsoft
Roanne Sones, Head of Windows Platform
Call to action

Session Objective(s):
Understand new scenarios possible with Windows 10 on ARM
Provide guidance on how to support Windows 10 on ARM
Rebuild drivers to support ARM64
The tools and documentation are ready
For More Information

https://aka.ms/Win10onARM

Contact woafeedback@moicrosoft.com if need more info.