DEEPX
DX-M1 & DX-M1M
NPU Accelerator
: Windows Quick User Manual

Revision: v1.1

Document History

Version	Date	Description	Author
V1.1	26.11.13	- Updated for DX-SDK Unified Installer v1.3.2 (installer UI screenshots and installation steps). Added firmware update procedure (Section 3.3.3). - Added DEEPX DXNN Benchmark (Section 3.3.5).	Brad Kim
V1.0	26.08.12	- Initial Release	Brad Kim JC Yoo

Copyright ⓒ 2026 by DEEPX. All rights reserved.

TERMS OF USE

DEEPX Company Limited (DEEPX) reserves the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to DEEPX's terms and conditions of sale supplied at the time of order acknowledgment.

DEEPX assumes no liability for application assistance or customer product designs. Customers are responsible for their products and applications using DEEPX components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

DEEPX does not warrant or represent that any license, either express or implied, is granted under any DEEPX patent right, copyright, mask work right, or other DEEPX intellectual property right relating to any combination, machine, or process in which DEEPX products or services are used. Information published by DEEPX regarding third-party products or services does not imply any license, warranty, or endorsement by DEEPX. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from DEEPX under the patents or other intellectual property of DEEPX.

Reproduction of DEEPX information in DEEPX data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alterations is an unfair and deceptive business practice. DEEPX is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

DEEPX products are not authorized for use in safety-critical applications (such as life support) where a failure of the DEEPX product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of DEEPX products in such safety-critical applications, notwithstanding any application-related information or support that may be provided by DEEPX. Further, Buyers agree to indemnify DEEPX and its representatives from any claims or damages arising from unauthorized use of DEEPX products in safety-critical applications.

DEEPX products are neither designed nor intended for use in military/aerospace applications or environments unless the DEEPX products are specifically designated by DEEPX as military-grade or "enhanced plastic."

DEEPX products are neither designed nor intended for use in automotive applications or environments unless the specific DEEPX products are designated by DEEPX as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, DEEPX will not be responsible for any failure to meet such requirements.

ⓒ DEEPX Co., Ltd.

All rights reserved. No part of this document may be reproduced in any form without written permission from DEEPX Co., Ltd. DEEPX Co., Ltd. reserves the right to change in its products or product specifications to improve function or design at any time, without notice.
Address: 3F, 20, Pangyoyeok-ro 241beon-gil, Bundang-gu, Seongnam-si, Gyeonggi-do, Republic of Korea
Tel : +82-31-603-9877
URL : http://www.deepx.ai
Email : tech_support@deepx.ai

Contents

1. Introduction 5

2. System Overview 6

2.1. Hardware Architecture: DX-M1 M.2 6

3. Getting Started 7

3.1 System Prerequisites 7

3.2 Software Setup and Installation 8

3.2.1. [Step 1] Download & Prerequisites Check 8

3.2.2. [Step 2] Configure Setup Wizard 9

3.2.3. [Step 3] Installation & Complete 15

3.3. Desktop Management Tools & Operations 17

3.3.1. Overview & Desktop Shortcuts 17

3.3.2. Executing Demo Applications (DEEPX Demo Launcher) 18

3.3.3. Updating Firmware (DEEPX Firmware Downloader) 19

3.3.4. Real-Time System Monitoring (DX Top Monitor) 21

3.3.5. Benchmarking Model Performance (DEEPX DXNN Benchmark) 22

3.4. CLI Tools 23

3.4.1. Model Inspection Tool (dxparse) 23

3.4.2. Run Model (dxrun) 24

3.4.3. DX-RT CLI Tool (Firmware Interface) (dxcli) 25

3.4.4. NPU Monitoring Utility (dxtop) 26

3.4.5. Performance Benchmarking Utility (dxbenchmark) 28

Copyright ⓒ 2026 by DEEPX. All rights reserved.

1. Introduction

The **DEEPX DX-M1 (& DX-M1M)** is an entry-to-mid edge AI inference accelerator designed for high-performance neural network workloads. This Quick User Manual serves as a technical guide for installing and configuring the DX-M1 M.2 Accelerator Card on x86-64 Windows systems.

Key topics covered in this manual include:

- **Hardware Architecture & Specifications (Chapter 2):** Overview of the DX-M1 M.2 form factor, PCIe Gen3 x4 interface, thermal characteristics, and hardware performance.
- **Software Setup & Installation (Chapter 3.1 ~ 3.2):** System prerequisites, .NET Desktop Runtime checks, and guided deployment of the Windows driver, DX-RT runtime service (`dxrtd`), and SDKs.
- **Management Tools & Operations (Chapter 3.3):** Operational guides for running pre-built AI inference demos (DEEPX Demo Launcher), updating NPU firmware (DEEPX Firmware Downloader), telemetry monitoring (DX Top Monitor), and model performance benchmarking (DEEPX DXNN Benchmark).

Copyright ⓒ 2026 by DEEPX. All rights reserved.

2. System Overview

This chapter provides a technical overview of the **DEEPX DX-M1** accelerator and its integration environment.

2.1. Hardware Architecture: DX-M1 M.2

The system utilizes a high-performance heterogeneous architecture specifically engineered to support demanding edge AI workloads.

DX-M1 M.2 Accelerator Card Specifications

The **DX-M1** is a high-efficiency AI inference accelerator designed for seamless integration into edge computing environments through its optimized internal architecture.



Figure 1. DX-M1 M.2 Module

Table 1. DX-M1 M.2 Technical Specifications

Features	Details
AI Performance	25 TOPS (INT8)
Host Interface	PCIe Gen3 x4 (Supports Gen 1/2/3 & x1/x2/x4)
Memory	4GB LPDDR5 (5600 MT/s)
Power Consumption	1W (min) ~ 5W (max)
Operating Temperature	0 ~ 70°C (Commercial)
	-25 ~ 85°C (Extended)
	-40 ~ 85°C (Industrial)
Thermal Solution	Heatsink (Option)
Form Factor	M.2 2280(M Key), 22 x 80mm
OS Support	Windows 11/10
	Debian-based Linux (Ubuntu 26.04/24.04/22.04/20.04 LTS)
	Yocto Project and Docker
	Android
AI Frameworks	Ultralytics, PyTorch, TensorFlow, ONNX, Keras
System Support	x86 and ARM-based Architectures

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3. Getting Started

This chapter provides a concise guide to configuring the **DEEPX DX-M1** development environment. It covers the essential steps from verifying system requirements to executing your first AI inference benchmarks.

3.1 System Prerequisites

Before installing the DX-M1 drivers and software, ensure your system meets the following minimum requirements:

Table 2. System Prerequisites

Item	Requirement
Operating System	**Windows 10** (version 22H2 or later, 64-bit) or **Windows 11** (64-bit)
.NET Runtime	**NET Desktop Runtime 10.0.x (x64)** (An internet connection is **required** to download and install the runtime if it is not already installed.)
Administrator Privileges	**Administrator privileges** (**Required** to install the driver and software)
Python	**Python 3.10–3.14** (Optional: required only if installing the Python SDK)

Note: Please ensure that all pending Windows updates are completed and you are logged in with administrator rights before proceeding with the driver installation.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.2 Software Setup and Installation

This section guides you through installing the DEEPX SDK, drivers, and runtime using the unified Windows setup wizard (.exe). The installer sequentially handles system diagnostics, component selection, automated deployment, and post-installation verification to prepare your system for AI inference benchmarking.

3.2.1. [Step 1] Download & Prerequisites Check

Prior to launching the setup wizard, verify system prerequisites and download the latest installer package.

[Step 1-1] Prerequisite Check (.NET Desktop Runtime 10):

Before running the SDK installer, verify whether **.NET Desktop Runtime 10 (x64)** is installed on your system. If it is not installed, download and install it prior to proceeding.

- **Download:** .NET Desktop Runtime 10 (x64) installer
  https://dotnet.microsoft.com/download/dotnet/thank-you/runtime-desktop-10.0.10-windows-x64-installer

[Step 1-2] Download and Run Installer

(1) Visit the DEEPX Developer Portal to download the latest DX-M1 Windows SDK installer.

- **Download Portal**: DEEPX Developer Portal
  https://developer.deepx.ai/sw-download/

(2) Locate the downloaded installer executable (.exe), right-click the file, and select **Run as administrator**.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.2.2. [Step 2] Configure Setup Wizard

Launch the setup wizard and configure the installation options across the following guided screens:

[Step 2-1] Select SDK Package

When the setup wizard launches, the **Select SDK Packages** screen will appear. Select the package that corresponds to your hardware configuration, then click **Next**.
Note: For **DX-M1** hardware, select **DX-M1/M1M/H1 SDK**.

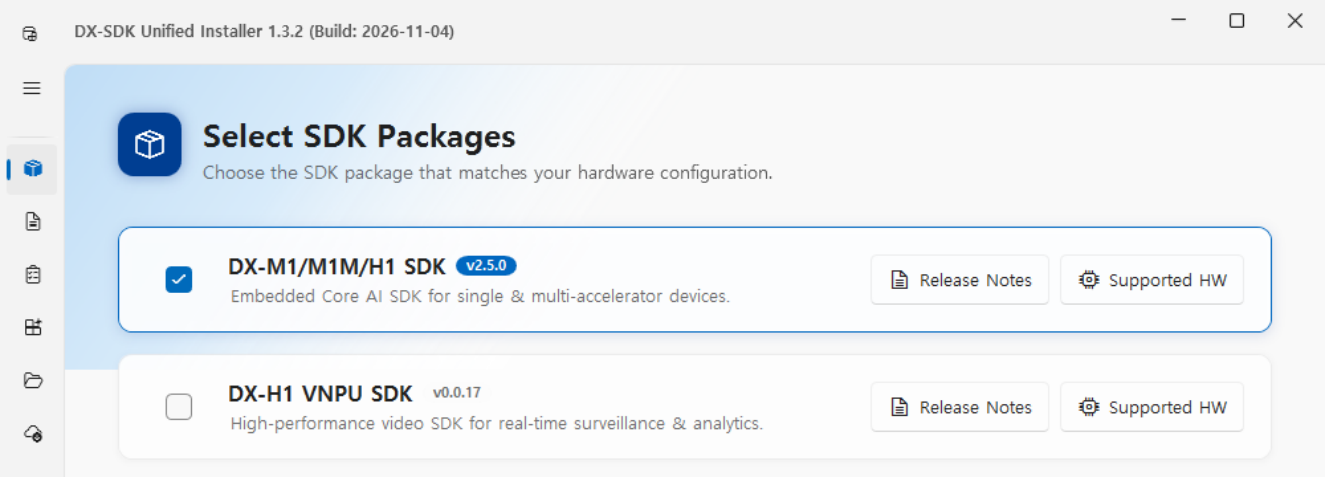


Figure 2. Select SDK Package

[Step 2-2] Accept the License Agreement

When the **License Agreement** screen appears, carefully read the End User License Agreement (EULA). After reviewing the agreement, **check if I accept the license agreement**, then click **Next** to continue.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

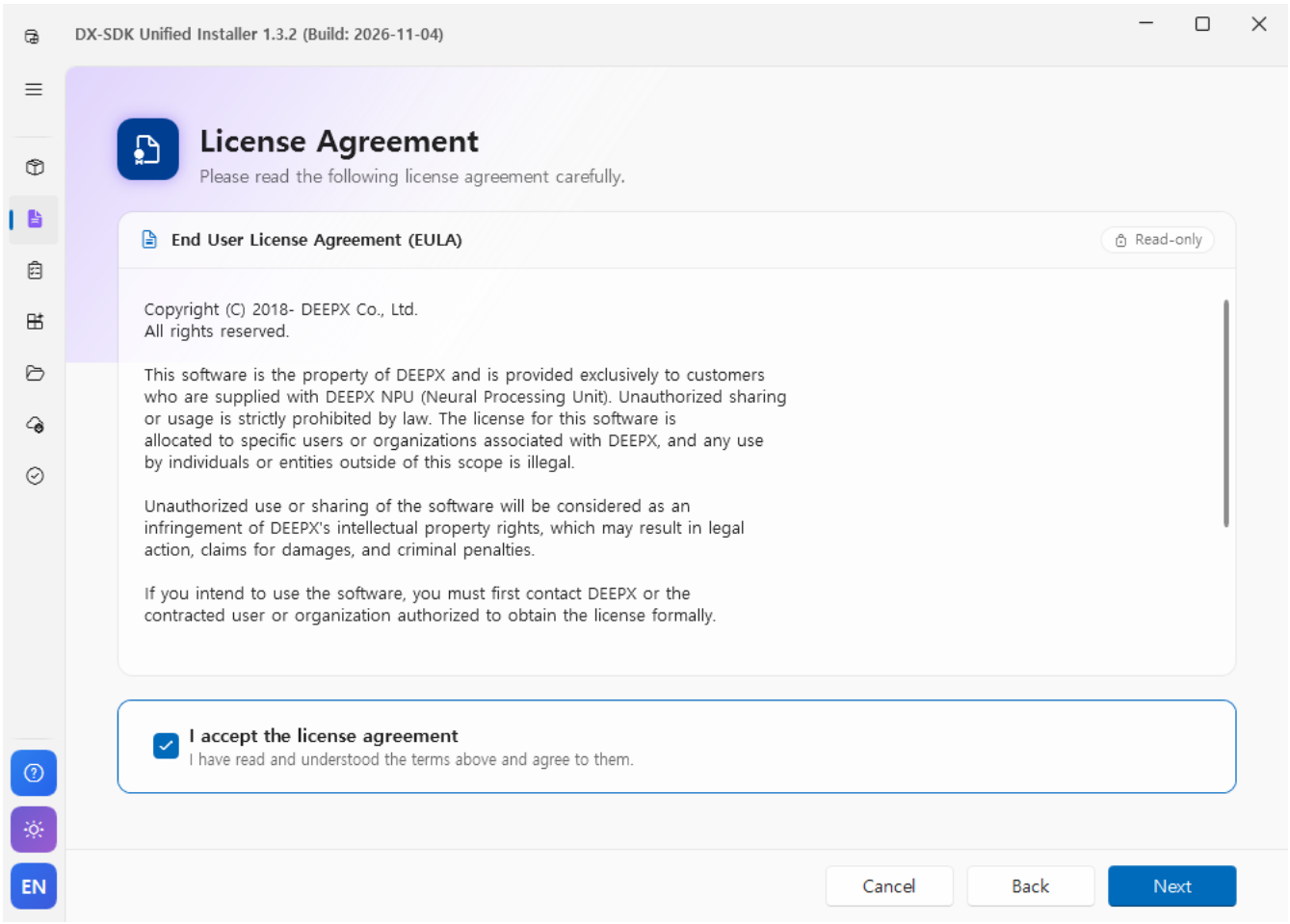


Figure 3. Accept the License Agreement

Copyright ⓒ 2026 by DEEPX. All rights reserved.

[Step 2-3] System Requirements Check

The installer automatically evaluates your system's hardware and software environment across two main categories:

- **COMMON Section:** Verifies Administrator privileges, Operating System version, .NET Desktop Runtime, VC++ Redistributable, and Python installation.
- **DX-M1 Section:** Checks the recognition status of the DX-M1 hardware, drivers, and background service environment.

Note on Python: Python is required only if you plan to install the Python SDK component (supported versions: **3.10–3.14**). If Python is already installed on your PC, the screen displays the default Python version currently detected. If you do not use the Python SDK, you can proceed without Python.

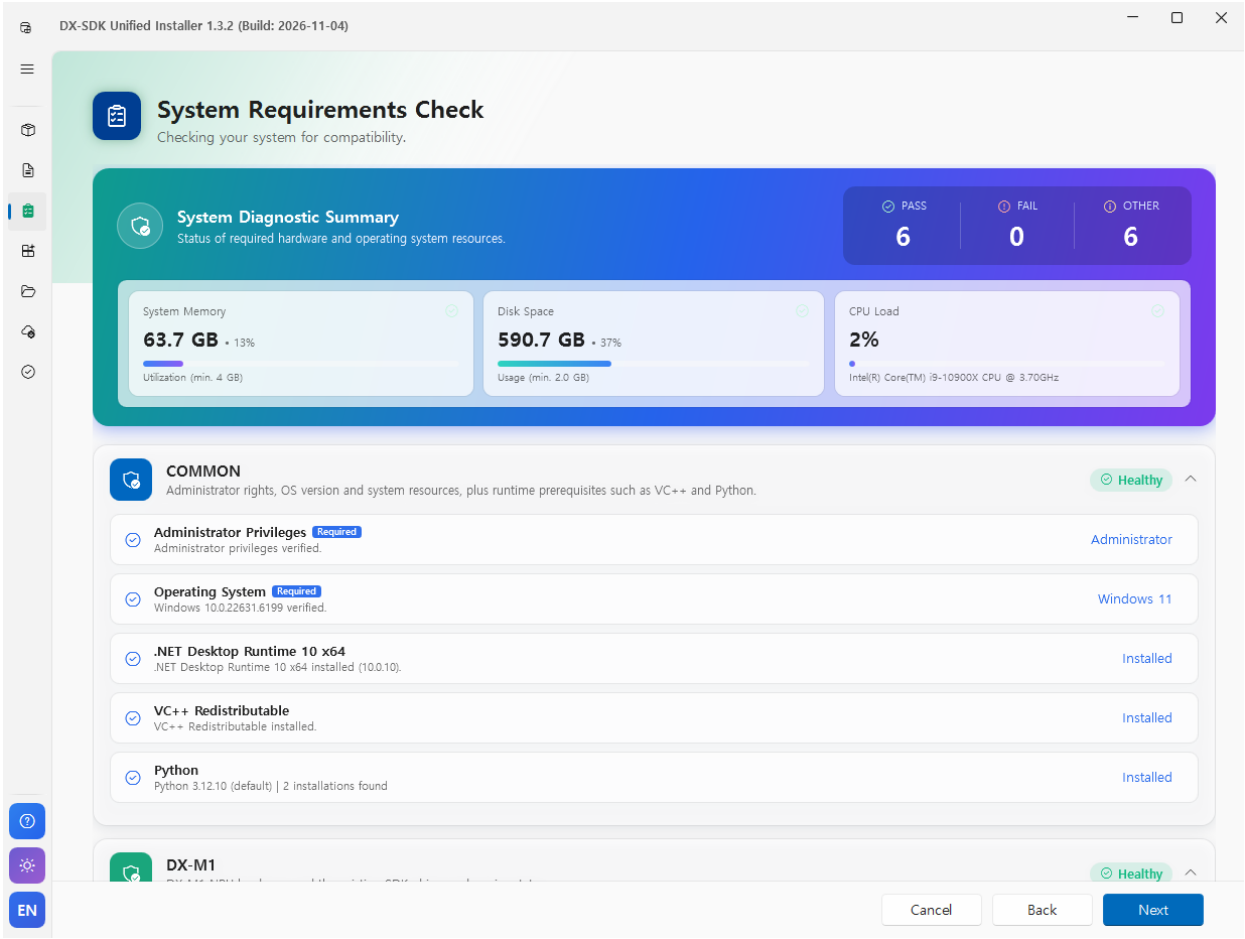


Figure 4. System Requirements Check - System Diagnostics

On a first-time installation, it is normal for the DX-M1 driver/service items to display as **To be installed**, as shown below.

Copyright ⓒ 2026 by DEEPX. All rights reserved.



Figure 5. System Requirements Check - Hardware & Service Recognition

Once all mandatory checks pass, click **Next**.

[Step 2-4] Select Components

Select the components you wish to install. You can choose a predefined installation profile (**Minimal**, **Standard**, **Full**, or **Custom**).
Choosing **Full** includes all of the following components:

- **DEEPX M1 Driver (Required):** Core device driver for the DEEPX M1 NPU hardware.
- **DX-RT Runtime (Required):** Includes the runtime service (`dxrtd`) for resource scheduling and monitoring, alongside developer tools (`dxcli`, `dxtop`).
- **C/C++ SDK:** Headers, libraries, and CMake configuration for native C/C++ development.
- **Python SDK:** Python wheel (`dx_engine`) supporting Python 3.10–3.14.
- **Demo Applications:** Pre-built inference demos (e.g., YOLO, EfficientNet, DeepLab, SCRFD).

Note: Mandatory components (Driver and Runtime) are checked by default and cannot be deselected.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

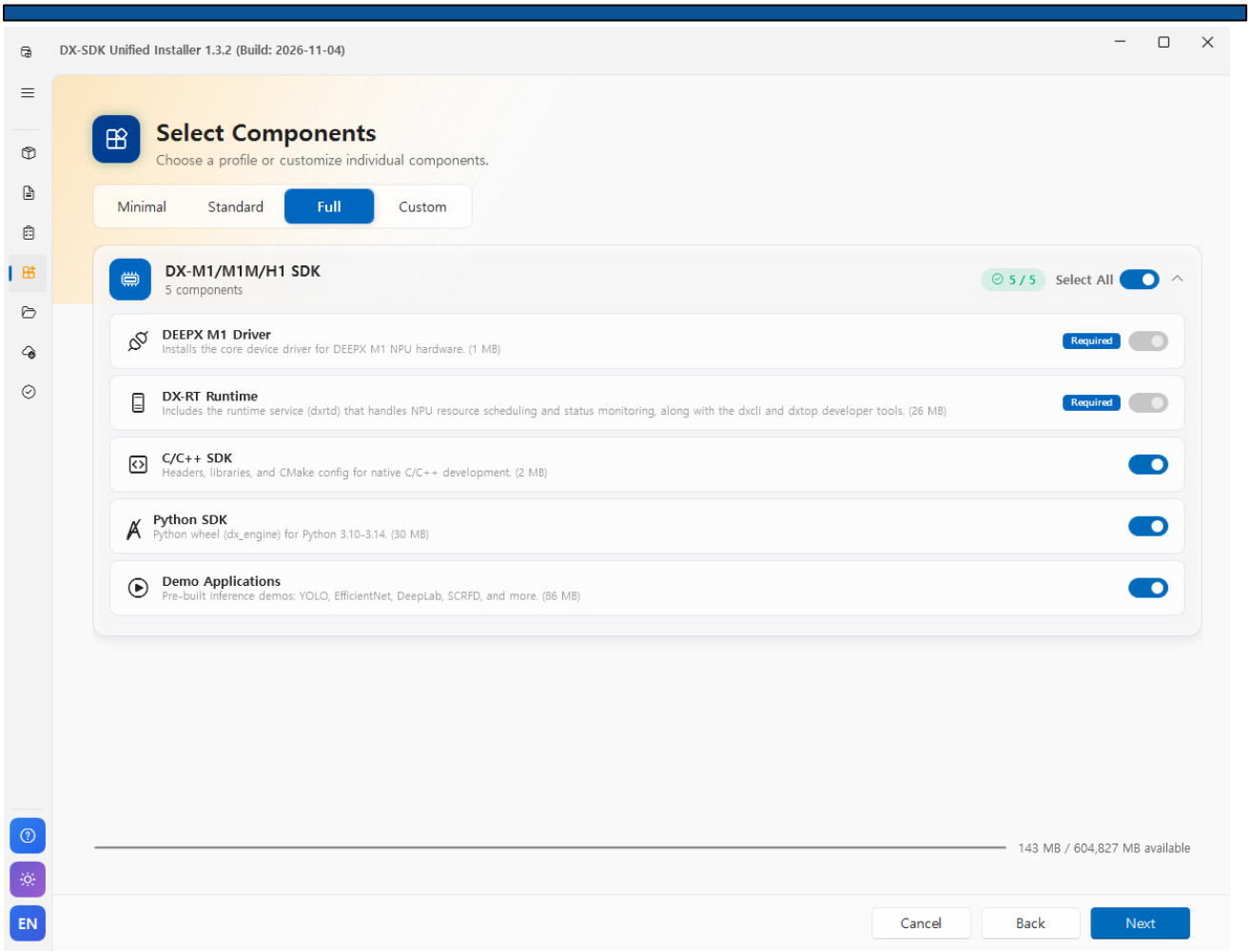


Figure 6. Select Components

After selecting your desired components, click **Next**.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

[Step 2-5] Choose Install Location

Confirm the base folder for the installation. Each product is installed in its own folder inside the base folder, as shown in the **Installation layout** section:

- **Default Base Folder:** C:\Program Files\DEEPX
- **DX-M1/M1M/H1 SDK Installation Path:** C:\Program Files\DEEPX\M1_M1M_Series\DX-RUNTIME\2.5.0
- To specify a different base folder, click **Browse...** to select your desired target folder.

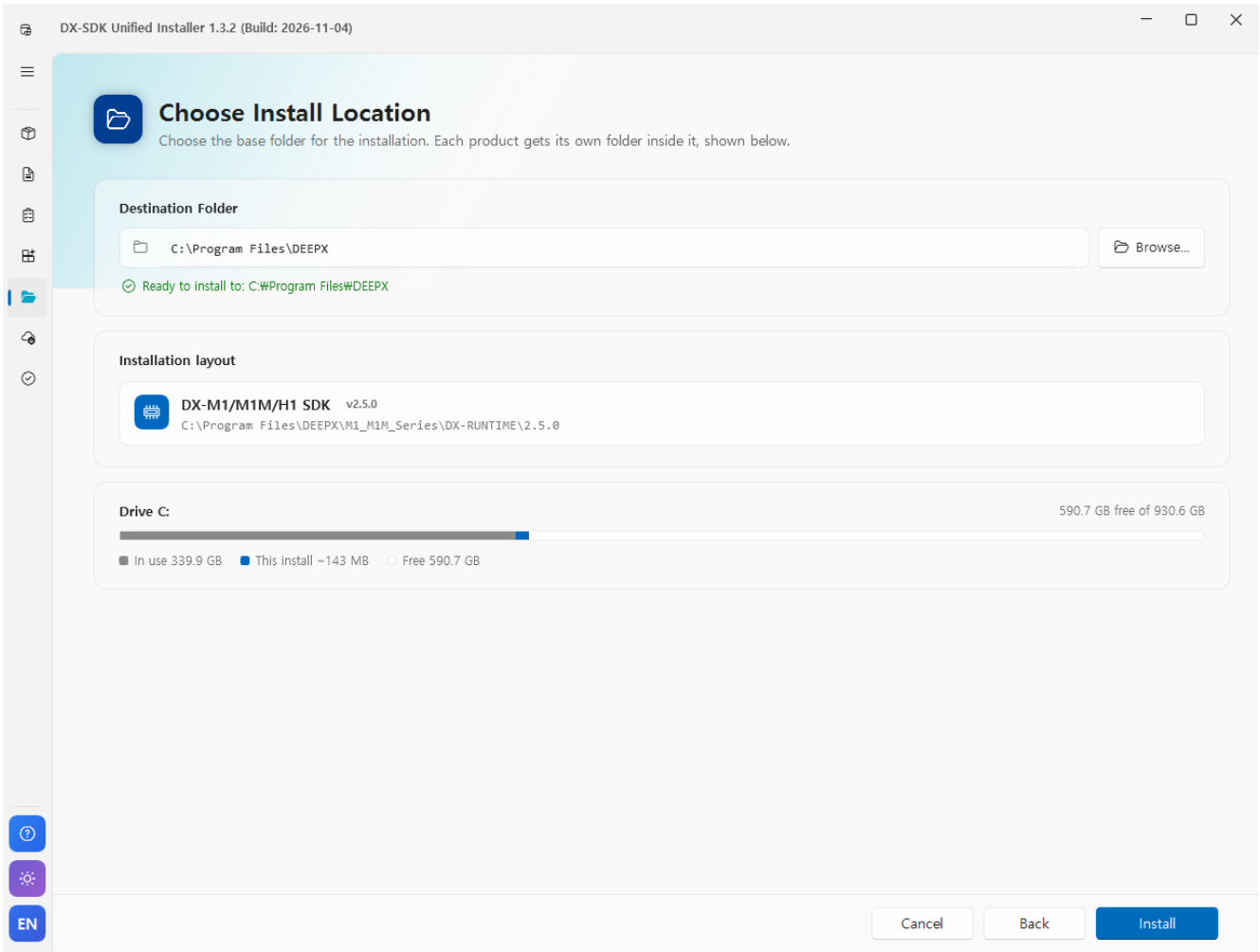


Figure 7. Choose Install Location

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.2.3. [Step 3] Installation & Complete

Monitor the automated setup progress and review the post-installation quick-access options upon completion.

[Step 3-1] Progress the Installation

Click **Install** to begin the installation. The setup wizard will display the real-time installation progress as components are extracted and configured on your system.

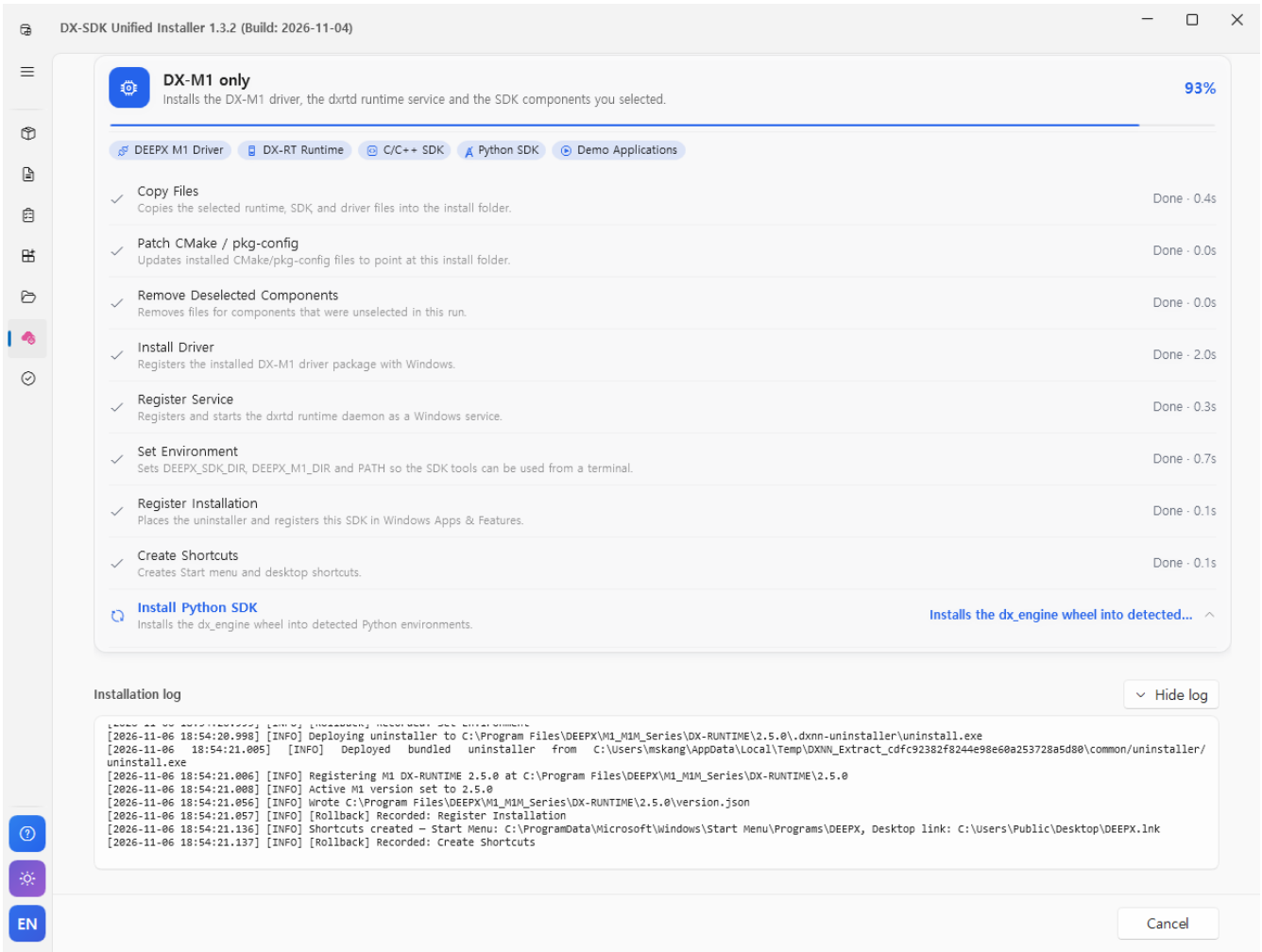


Figure 8. Progress the Installation

Copyright ⓒ 2026 by DEEPX. All rights reserved.

[Step 3-2] Complete the Installation

Once the installation finishes, the **Installation Complete** screen appears, providing the following information and quick-access options:

- **Restart required:** Windows needs to restart to finish installing the drivers. The device may not be detected until Windows is restarted.
- **Verification Result:** Displays the number of verified items (e.g., 6 verified). Click **Details** for more information.
- **Firmware:** Displays the firmware version bundled with this setup and the current firmware version of each detected device. If the device firmware is older than the bundled version, **Update needed** is displayed.
- **Open DEEPX Launcher:** Launches the DEEPX Launcher GUI to run SDK tools and demos as cards.
- **User Guide:** Opens installation, firmware update, and troubleshooting documentation in your default web browser.

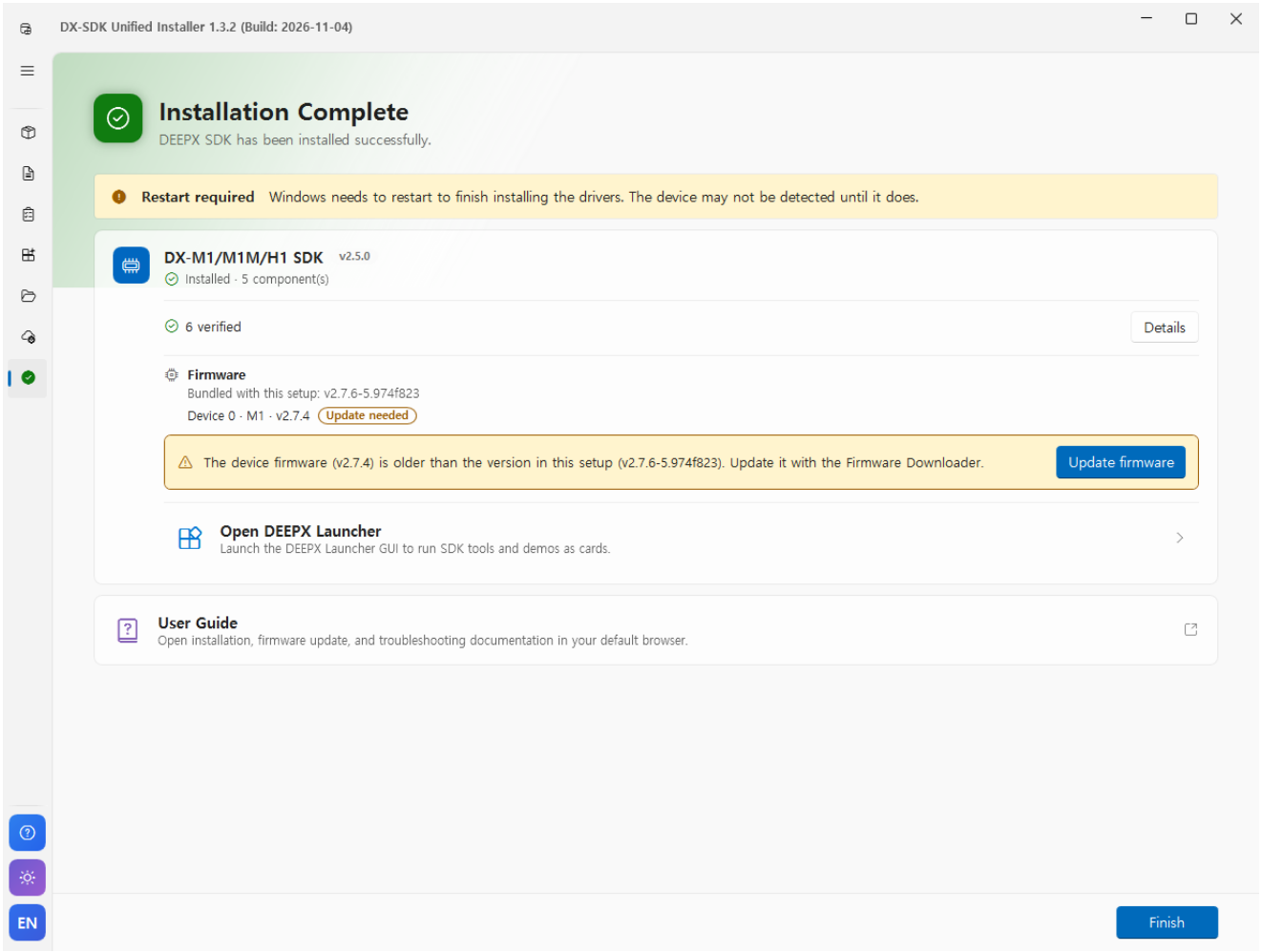


Figure 9. Installation Complete Summary

After reviewing the installation summary, click **Finish** to close the setup wizard, and then restart Windows.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.3. Desktop Management Tools & Operations

This section provides an overview of the desktop management applications created during setup, along with detailed instructions for running AI inference demos, updating NPU firmware, monitoring real-time system telemetry, and benchmarking model performance.

3.3.1. Overview & Desktop Shortcuts

Upon successful installation, a **DEEPX** folder is automatically created on your desktop. This folder contains shortcuts to primary management utilities used for device verification and maintenance.

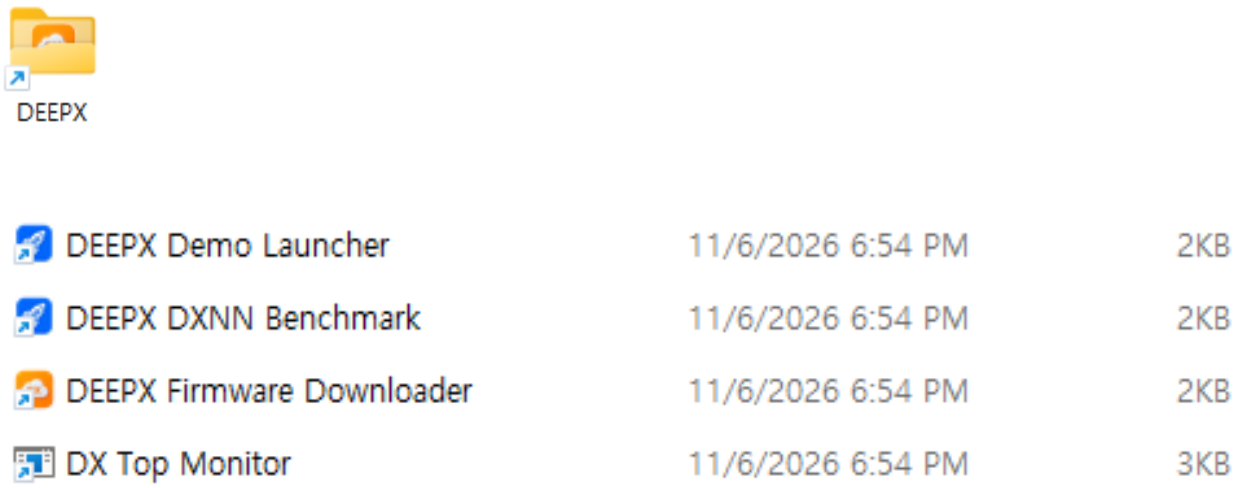


Figure 10. DEEPX Desktop Folder Shortcuts

To launch any utility, open the **DEEPX** desktop folder and double-click the corresponding application icon:

- **DEEPX Demo Launcher:** Runs pre-compiled AI inference demos to verify NPU operation.
- **DEEPX DXNN Benchmark:** Measures the inference performance (FPS) of DXNN models on the NPU and compares the results with official FPS data.
- **DEEPX Firmware Downloader:** Inspects current NPU firmware details and applies firmware updates.
- **DX Top Monitor:** Displays real-time hardware telemetry, workload, and operational status.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.3.2. Executing Demo Applications (DEEPX Demo Launcher)

Open the **DEEPX** desktop folder and double-click **DEEPX Demo Launcher** to start the interactive GUI demo suite.

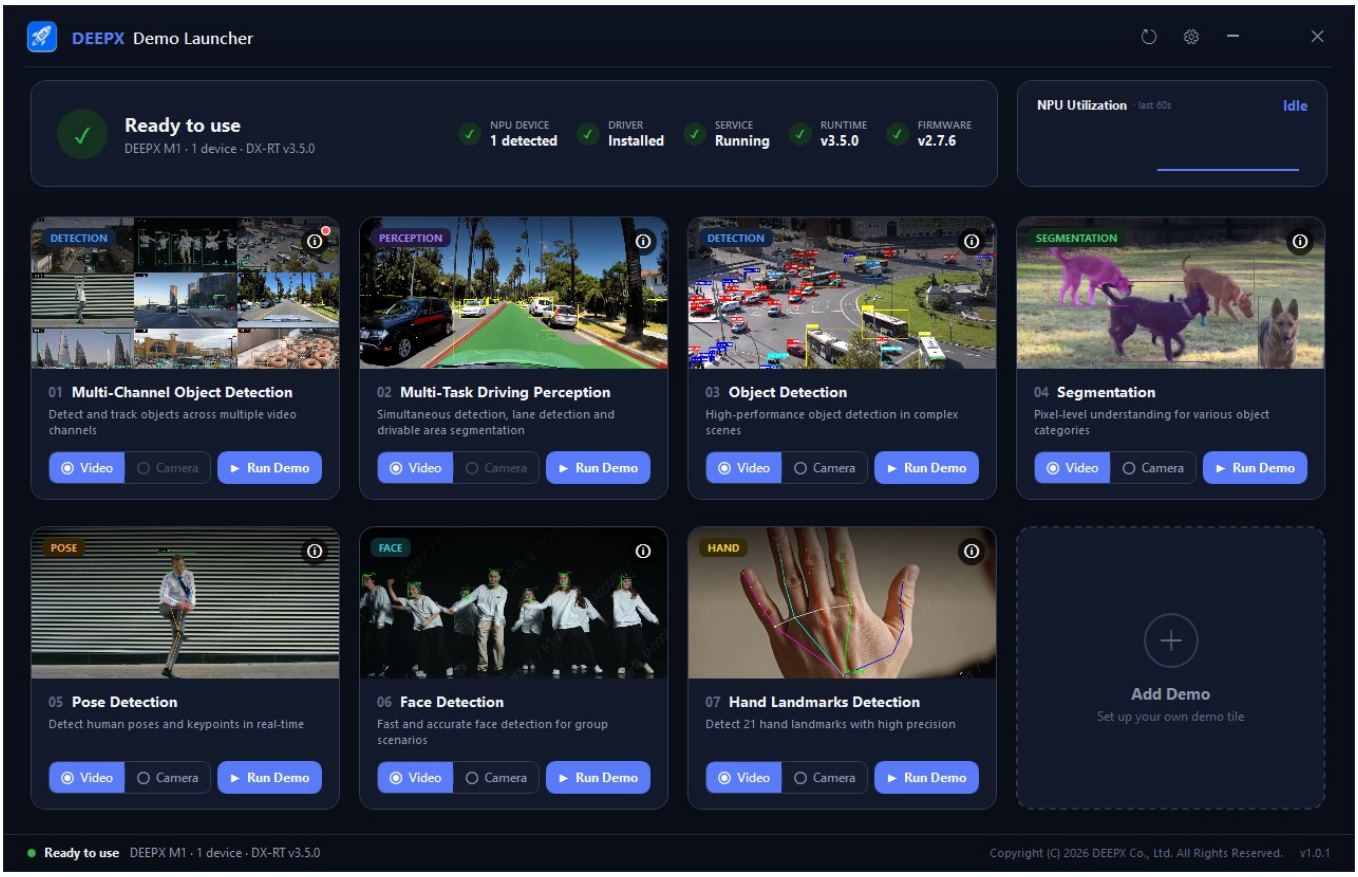


Figure 11. DEEPX Demo Launcher Interface

The Demo Launcher includes a variety of pre-packaged AI inference workloads designed to showcase NPU performance:

Table 3. Pre-built Demo Applications

No.	Demo Name	Description
01	Multi-Channel Object Detection	Detects and tracks objects simultaneously across multiple video streams.
02	Multi-Task Driving Perception	Performs simultaneous object detection, lane detection, and drivable area segmentation.
03	Object Detection	Delivers high-performance object detection in complex, real-world scenes.
04	Segmentation	Executes pixel-level semantic segmentation for various object categories.
05	Pose Detection	Estimates human body poses and keypoints in real time.
06	Face Detection	Provides fast and precise face detection optimized for multi-face scenarios.
07	Hand Landmarks Detection	Detects 21 3D hand landmarks with high precision.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.3.3. Updating Firmware (DEEPX Firmware Downloader)

Open the **DEEPX** desktop folder and double-click **DEEPX Firmware Downloader** to inspect current NPU firmware details and update the firmware to the latest release if necessary.

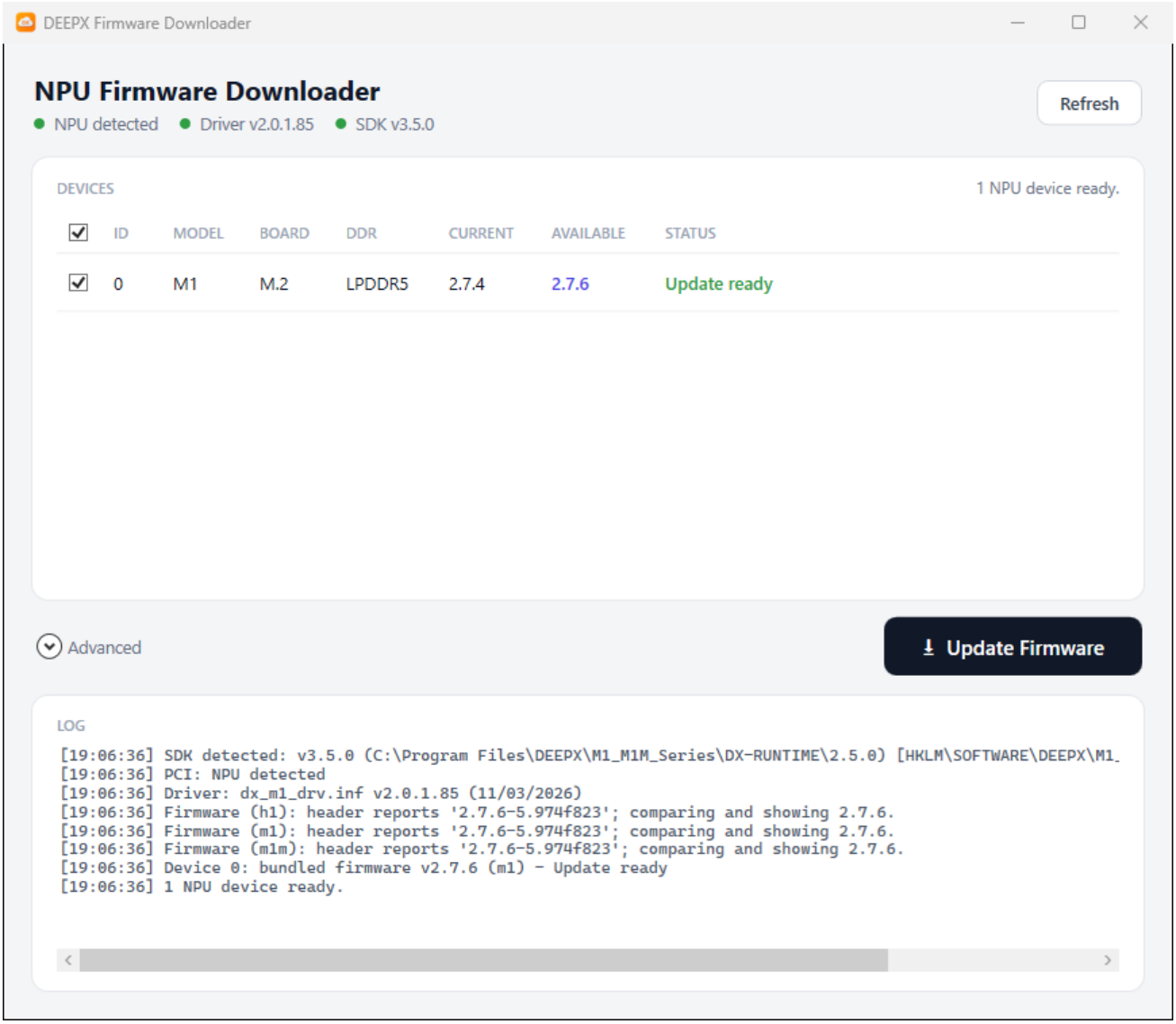


Figure 12. DEEPX Firmware Downloader Interface

To update the firmware, follow these steps:
(1) In the **DEVICES** list, check the **CURRENT** and **AVAILABLE** firmware versions of each device. If a newer firmware version is available, the **STATUS** displays **Update ready**.
(2) Select the checkbox of the device(s) to update, and then click **Update Firmware**.
(3) During the update, the device reboots automatically and is detected again once the reboot is complete. Once the update completes, the **STATUS** changes to **Updated** and the **CURRENT** version matches the **AVAILABLE** version, as shown below.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

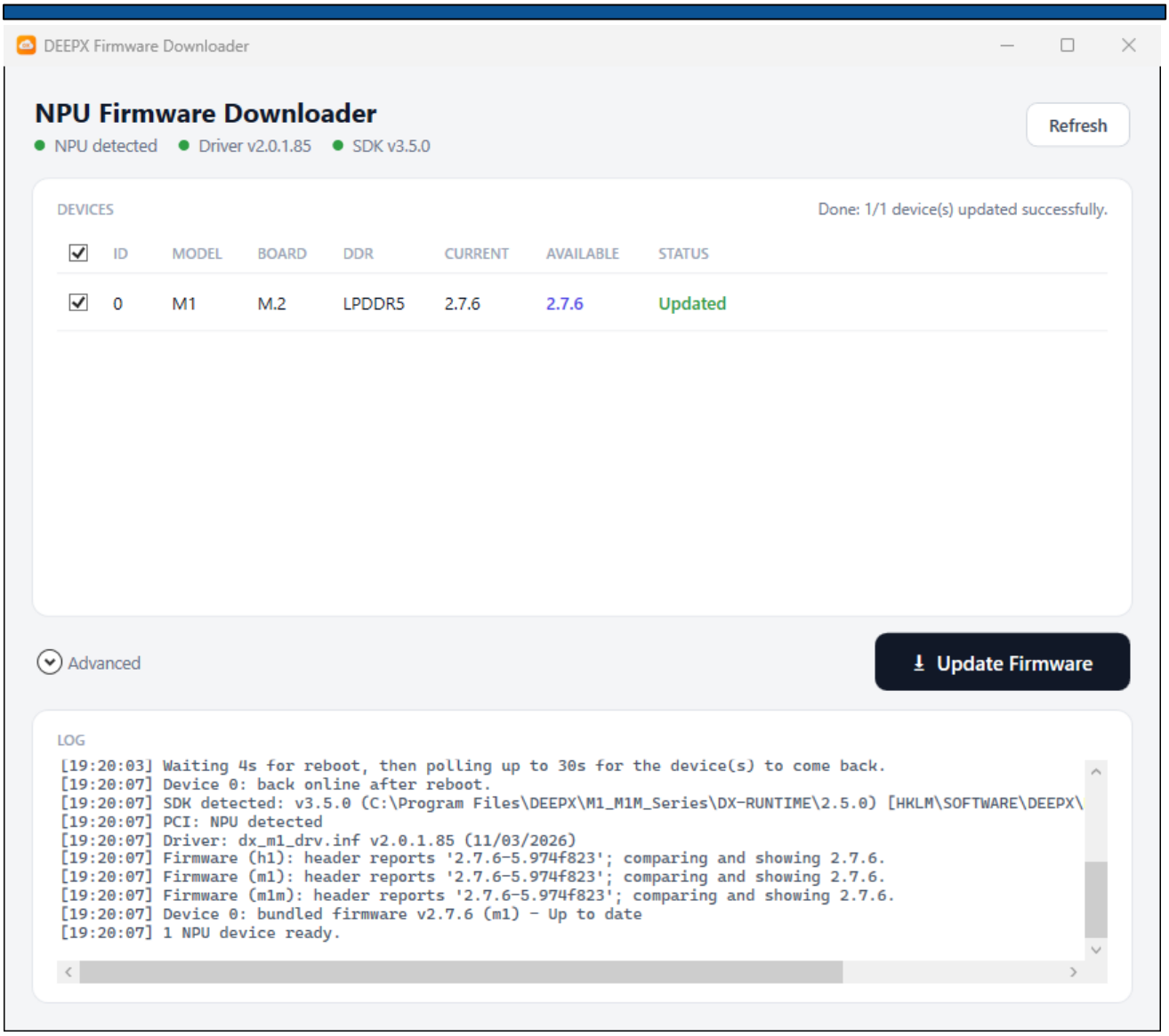


Figure 13. Firmware Update Complete

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.3.4. Real-Time System Monitoring (DX Top Monitor)

Open the **DEEPX** desktop folder and double-click **DX Top Monitor** to launch the real-time telemetry dashboard. This utility provides live visibility into device recognition, operating temperature, power consumption, and overall NPU utilization metrics.

Note: Starting from firmware v2.7.6, 0 mV is displayed in the **Voltage** field while the NPU is idle.

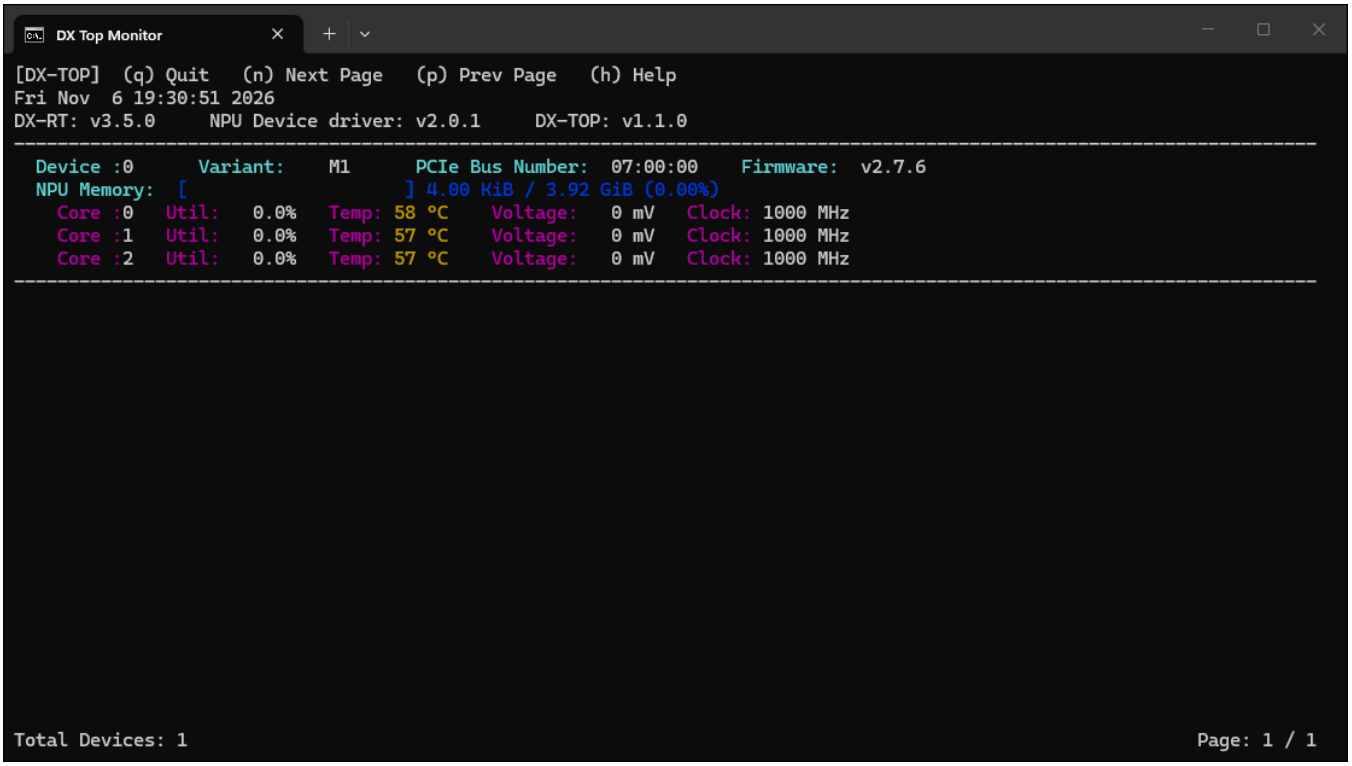


Figure 14. DX Top System Monitor Interface

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.3.5. Benchmarking Model Performance (DEEPX DXNN Benchmark)

Open the **DEEPX** desktop folder and double-click **DEEPX DXNN Benchmark** to measure the inference performance (FPS) of DXNN models on the NPU and compare the measured results with official FPS data.

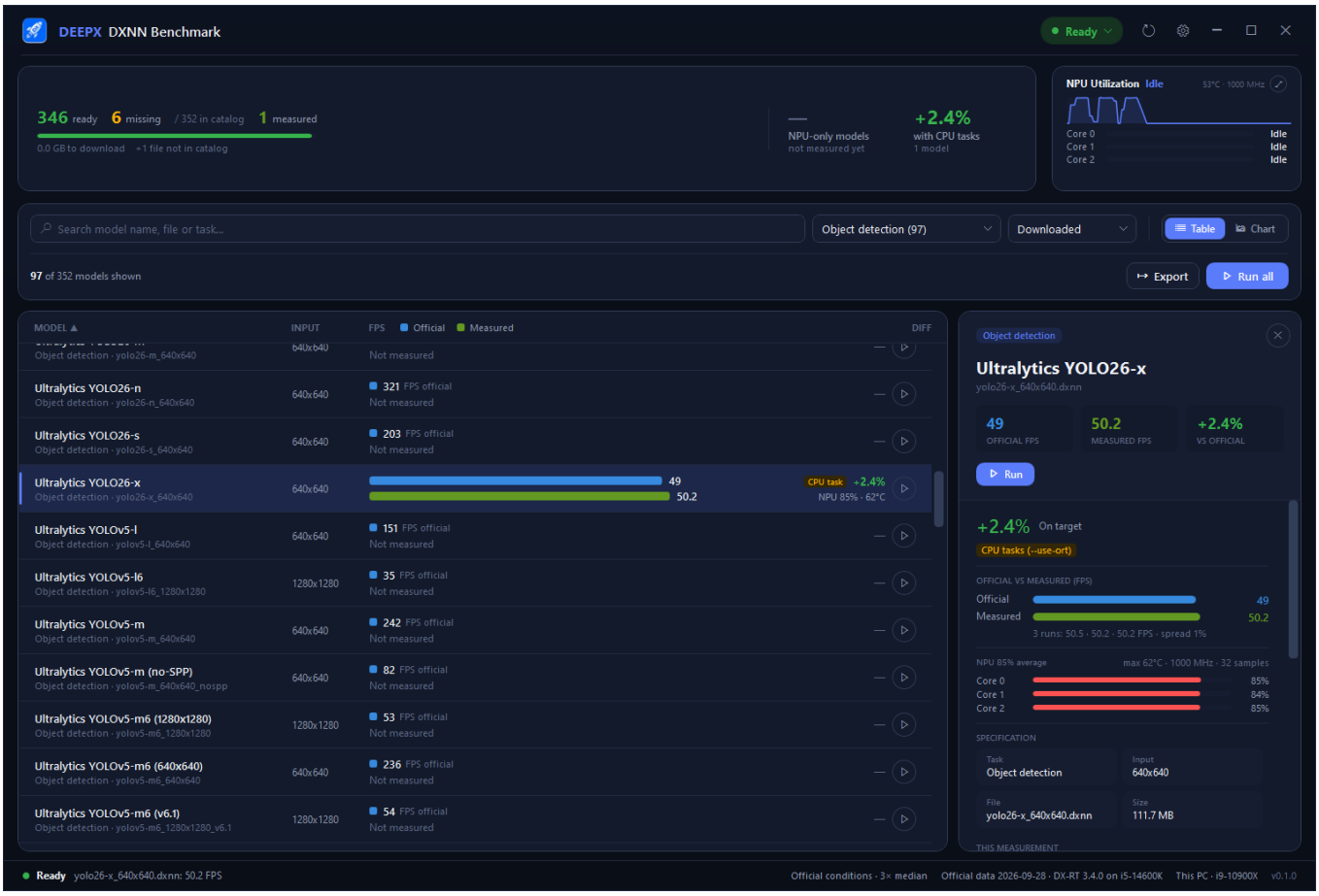


Figure 15. DEEPX DXNN Benchmark Interface

The main screen provides the following features:

- **Model Summary:** Displays the number of models that are ready, missing, and measured out of the models in the catalog, along with the average FPS difference for NPU-only models and for models with CPU tasks.
- **NPU Utilization:** Displays the real-time utilization of each NPU core, along with the temperature and clock.
- **Search and Filter:** Search models by model name, file, or task, and filter the model list using the drop-down menus. Click **Table** or **Chart** to switch the view.
- **FPS:** Displays the **Official** FPS and the **Measured** FPS on this PC for each model.
- **DIFF:** Displays the difference between the measured FPS and the official FPS as a percentage. Models that include CPU tasks are marked with a **CPU task** label.
- **Run all:** Runs the benchmark for all models shown in the list. While the benchmark is running, the button changes to **Stop**.
- **Model Details:** Displays the details of the selected model, including the official FPS, measured FPS, difference from the official FPS, the FPS of each run, average utilization of each NPU core, and model specifications (task, input size, file, and size). Click **Run** to run the benchmark for the selected model only.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.4. CLI Tools

This section introduces the **DX-RT** command-line tools for model inspection, inference testing, and device management. These tools allow developers to validate models, measure performance, and control hardware directly from the terminal.

These utilities are essential for validating model compilation, tuning inference performance, and maintaining the health and configuration of the underlying NPU hardware.

Legacy Command Aliases

The legacy command names from earlier releases are preserved as aliases and continue to work:

New Command	Legacy Alias	Description
dxparse	parse_model	Inspects compiled .dxnn model structure and metadata
dxrun	run_model	Executes inference benchmarks and stress tests
dxcli	dxrt-cli	Manages hardware resets, status queries, and firmware updates

3.4.1. Model Inspection Tool (dxparse)

This tool parses a compiled .dxnn model file and displays detailed information such as model structure, input/output tensors, and metadata.

Basic Usage

```
CMD

dxparse -m <model_path>
```

Command Options:

```
Plaintext

 -m,  --model FILE   Model file path (.dxnn) [Required]
 -v,  --verbose      Show detailed task dependencies and memory allocation
 -o,  --output FILE  Save console output to a file (without color codes)
 -j,  --json         Extract JSON binary graph data (graph_info, rmap_info)
 -h,  --help         Display help message
```

Example

```
CMD

dxparse -m model.dxnn
```

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.4.2. Run Model (dxrun)

This tool executes a compiled model to verify basic functionality, measure inference time, and optionally perform repeated runs for stress testing.

Basic Usage

```
CMD

dxrun -m <model_path> [-l <number of loops>] [-t <seconds>] [-d <devices>] [--use-ort] [--profiler] [--buffer-count <N>]
```

Command Options

```
Plaintext

-m,   --model FILE       Model file path (.dxnn) [Required]
-b,   --benchmark        Run maximum throughput benchmark (Default mode)
-s,    --single          Run single-input sequential inference test
-v,   --verbose          Display detailed NPU processing time and latency
-l,    --loops N         Number of inference loops to execute (Default: 30)
-t,    --time N          Inference duration in seconds (Overrides --loops)
-w,   --warmup-runs N    Number of warmup iterations before timing (Default: 0)
-d,   --devices ARG      Target NPU devices ('all', '0', '0,1', or 'count:N')
      --buffer-count N   Set input/output buffer pipeline count (Range: 1–100, Default: 6)
      --profiler         Enable profiler and export profiler.json
      --use-ort          Enable ONNX Runtime for CPU tasks in the model graph
-h,   --help             Display help message
```

Example

```
CMD

:: Run basic inference benchmark for 100 loops
dxrun -m model.dxnn -l 100

:: Timed benchmark test for 60 seconds with verbose latency output
dxrun -m model.dxnn --benchmark -t 60 -v

:: Enable profiling and pipeline buffer tuning
dxrun -m model.dxnn --profiler --buffer-count 10 -l 200
```

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.4.3. DX-RT CLI Tool (Firmware Interface) (dxcli)

This tool provides a CLI for interacting with **DX-RT** accelerator devices, supporting device status queries, hardware resets, and firmware updates.

Note: This tool is applicable **only** for accelerator devices.

Basic Usage

```
CMD

dxcli <option> [arguments]
```

Command Options:

```
Plaintext

-s,   --status              Display overall device status
-i,   --info                Display detailed hardware information
-m,   --monitor SEC         Monitor device status continuously every N seconds
-r,   --reset [TYPE]        Reset device (0: NPU reset, 1: Full reset)
-d,   --device ID           Specify target device ID (Default: -1 for all)
-u,   --fwupdate FILE       Update NPU firmware image (.bin)
-g,   --fwversion FILE      Inspect firmware version embedded in a .bin file
-h,   --help                Display help message
```

Example

```
CMD

:: Query overall device status
dxcli -s
dxcli --status

:: Reset hardware (0: NPU reset, 1: Full reset)
dxcli -r
dxcli --reset=1

:: Update firmware image
dxcli -u fw.bin
dxcli --fwupdate fw.bin

:: Monitor device status continuously every 1 second
dxcli -m 1
```

Copyright ⓒ 2026 by DEEPX. All rights reserved.

Note on Firmware Updates in Windows:
If an issue occurs during a firmware update with `dxcli -u` or `dxcli --fwupdate`, temporarily stop the background service using Administrator privileges before retrying:

Example

```
CMD

:: Stop DX-RT background service (Run CMD as Administrator)
net stop dxrtd

:: Perform firmware update
dxcli -u C:\firmware\dx_m1_fw.bin

:: Restart DX-RT background service
net start dxrtd
```

3.4.4. NPU Monitoring Utility (dxtop)

DX-TOP is a real-time command-line utility designed to monitor the status, utilization, and health metrics of Neural Processing Unit (NPU) devices managed by the DX-Runtime. It provides critical information necessary for performance tuning, debugging, and system health checks.

Utility Versions (Header Information)
The top section of the screen displays the version of the core software components:

Item	Description
DX-RT	**DX-Runtime Version.** The version of the core software stack that manages NPU operations.
NPU Device Driver	**NPU Driver Version.** The version of the operating system driver responsible for hardware communication.
DX-TOP	**Monitoring Tool Version.** The version of the DX-TOP utility itself.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

Device Status Monitoring

DX-TOP organizes information by individual NPU devices installed in the system:

- **Device Hardware Information:** Identifies physical hardware and firmware configuration.

Item	Description
Device ID	Unique identifier assigned to the NPU card (e.g., Device :0).
Variant	Model or variant of the NPU hardware (e.g., M1).
PCIE Bus Number	PCI Express bus location of the device (e.g., 02:00:00).
Firmware	Embedded firmware version running on the NPU hardware.
NPU Memory	Memory utilization of onboard NPU memory (Used MiB / Total GiB and Usage %).

- **Core Performance Metrics:** Real-time operational metrics for each processing core.

Item	Description
Core ID	Identifier for individual processing core within the NPU (e.g., Core :0).
Util	**Utilization (Percentage).** Active core processing time percentage.
Temp	**Temperature (Degrees Celsius).** Current core operating temperature.
Voltage	**Voltage (Millivolts).** Electrical voltage supplied to the core.
Clock	**Clock Rate (Megahertz).** Current operating frequency of the core.

Command Controls (Help Menu)

The following keyboard commands are available in the DX-TOP interface:

Command	Action
(q) Quit	Exits the DX-TOP application.
(n) Next Page	Navigates to the next page of device information.
(p) Prev Page	Navigates to the previous page of device information.
(h) Help	Displays the help screen or summary of available commands.

Copyright ⓒ 2026 by DEEPX. All rights reserved.

3.4.5. Performance Benchmarking Utility (dxbenchmark)

The dxbenchmark tool is a CLI utility designed to automate the performance benchmarking of deep learning models and generate detailed, visualized reports:

- **Comprehensive Execution:** Runs all detectable models in the target directory, including subdirectories (via --recursive).
- **Detailed Reporting:** Generates a single, integrated HTML document with visualizations of key performance indicators (KPIs) and stability metrics.
- **Stability Analysis:** Includes time-series data visualization (Performance Metrics Over Loops) and Coefficient of Variation (CV) for jitter analysis.
- **Raw Data for Comparison:** Outputs version-tagged JSON/CSV files for comparison across different Runtime and Driver versions.

CLI Options & Usage

```
Plaintext

DXRT dxbenchmark
Usage:
 dxbenchmark [OPTION...]

    --dir arg             Model directory
    --result-path arg   Destination of result file (default: .)
    --sort arg            Sorting criteria (name: Model Name, fps: FPS, time: NPU Inference Time, latency: Latency)
                           (default: name)
    --order arg           Sorting order (asc: Ascending order, desc: Descending order) (default: asc)
 -l, --loops arg        Number of inference loops to perform (default: 0)
 -t, --time arg         Time duration to perform (default: 0)
    --warmup arg      Warmup time (default: 10)
 -n, --npu arg          NPU bounding (default: 0)
                        0: NPU_ALL, 1: NPU_0, 2: NPU_1, 3: NPU_2, 4: NPU_0/1, 5: NPU_1/2, 6: NPU_0/2
 -d, --devices arg      Specify target NPU devices ('all', '0', '0,1,2', 'count:N') (default: all)
    --only-data        Make only data result (csv, json)
    --recursive         Search models recursively in subdirectories (default: base directory only)
    --use-ort           Enable ONNX Runtime for CPU tasks in the model graph
 -v, --verbose          Shows NPU Processing Time and Latency
 -h, --help             Print usage
```

Copyright ⓒ 2026 by DEEPX. All rights reserved.

Basic Usage and Syntax

Execution requires specifying the model source directory and one of the termination conditions (`-t` or `-l`).

- **Mandatory Parameters:**

Execution Parameter	Format	Description
Model Directory	`--dir {Directory}`	Path to the directory containing models.
Termination Condition	`-t {Time}`	Time duration in seconds (Prioritized over `-l`).
Loop	`-l {Loops}`	Number of inference loops.

Note on Execution Termination Priority:

Execution duration (`-t {Time}`) takes precedence over loop count (`-l {Loops}`). Running based on time is generally recommended for benchmarking.

- **Execution Examples**

```
CMD

:: Run benchmark for all models in folder for 60 seconds each
dxbenchmark --dir C:\models\ -t 60

:: Recursive benchmark sorted by latency (descending) on target devices for 60 seconds each
dxbenchmark --dir C:\models\ --recursive --sort latency --order desc -d "0,1" -t 60
```

Output Files and Reporting

The tool generates a structured HTML report for visualization alongside raw data files:

- **HTML Report Structure and Visualization:**
  - **Filename Format:** `DXBENCHMARK_{YYYY_MM_DD_HHMMSS}.html`
  - **Example:** `DXBENCHMARK_2026_08_13_174251.html`
- **Raw Data Files:**
  - **Purpose:** Stores underlying data for version comparison and external analysis.
  - **Filename Format:** `rt_{Runtime Version}_dd_{Device Driver Version}_pd_{PCIe Driver Version}.json/csv`
  - **Example:** `rt_3.1.0_dd_1.7.1_pd_1.4.1.json`

Copyright ⓒ 2026 by DEEPX. All rights reserved.