**Table of Contents**

End-user license agreement (US) .................................................. 4
Important safety information .................................................... 5
  General safety instructions .................................................. 5
FCC RF exposure information .................................................. 5
  Cyborg operation ............................................................. 6
  For your safety .............................................................. 6
FCC part 15 class B compliance .................................................. 6
Robot ethics notice ............................................................... 6

1 Introduction and setup ......................................................... 7

2 Power overview ................................................................. 8
  Working with batteries .................................................... 8
  Power control ............................................................... 8
  Replenishing the battery .................................................. 9
  Low power ..................................................................... 10

3 Subsystems and their management .......................................... 11

4 Personas ............................................................................ 12
  What personas can do ..................................................... 12
  Creating new personas .................................................... 12
  Installing personas .......................................................... 13
  Extending the preconfigured message set ............................. 13

5 Applications ....................................................................... 14
  Creating new applications ................................................ 14

6 Devices and Peripherals ......................................................... 15
  Active devices ................................................................ 15
  Passive devices ................................................................ 15
  Creating new devices ........................................................ 15

7 Remote access ..................................................................... 16
  Connecting to a unit .......................................................... 16
  Scanning nearby units without connecting ........................... 16
  Sending a public announcement ........................................ 16
  Issuing a command to a single unit ...................................... 16
  Working with terminal mode .............................................. 17
  Accessing the TTY menu remotely ....................................... 17

Hiding and closing the remote management console ................. 17

8 Understanding what your DAX/2 experiences ......................... 18
  Cortex commands .......................................................... 18
  Cortex bypass commands .................................................. 18
  Understanding the vocoder pipeline .................................... 19
  Released speech ................................................................ 19
  Self access and local command execution ............................ 20

9 Managing packages and installed software ............................. 21

10 User management and access control .................................... 22
  Adding a new user ........................................................... 22
  Creating a manager .......................................................... 22
  Removing a user ............................................................... 22
  Transferring ownership ..................................................... 22
  Abandoned units .............................................................. 22
  Local access control ........................................................ 22
  Remote access control ...................................................... 23
  Self access control ........................................................... 23
  Locking ........................................................................... 23

11 Miscellaneous preferences ..................................................... 24

12 OEM data files ................................................................... 25

13 Sound .............................................................................. 26
  Voice notifications ............................................................ 26
  Chimes ............................................................................. 26
  Fan control ....................................................................... 26

14 Scripting actions with Arabesque ........................................... 27

15 Understanding the SXD software architecture ......................... 29
  Functions by component .................................................... 29
  Packages ........................................................................... 31

16 Troubleshooting ................................................................ 32

A1 Installing data files ............................................................. 34
A2 Command reference ........................................................... 35
By operating the hardware included in this package you agree to the following terms. If you do not agree to these terms, you are not permitted to install, use, or modify the DAX/2 system. If you reject these terms within fourteen (14) days of your purchase, you may contact Nanite Systems (hereafter “the company”) at 425-822-6008 to inquire about a full refund of the purchase price of the hardware. If you purchased the unit at retail, your right to return the hardware is subject to the retailer’s return policy.

Customization. You are allowed to modify or replace the housing of your controller at your leisure provided the circuit boards and other electronic components inside (other than the screen and external panel) are unaltered. If at the time of a warranty claim it is obvious that the controller’s circuitry has been damaged as a byproduct of customization, the company reserves the right to refuse maintenance or replacement.

Reverse engineering and piracy. The company makes every effort to provide a customizable, extensible, and well-documented platform for creative users and developers to enjoy. However, with the exception of certain modules based on GPL-licensed code, the firmware itself is not open source and is not meant to be used on hardware other than official Nanite Systems controllers or those sold by our partners under license. By using the DAX/2 system, you agree to not exploit undocumented internal functions, to not develop interoperable controller hardware that runs the SXD firmware, and to not collect or distribute any instrument or instructions to enable others to do so.

Software modification. You are permitted to extend, modify, and replace the firmware on your device for your own personal use, as well as to load any user applications onto the device for any reason. You agree not to hold the company responsible for any direct, indirect, consequential or special damages resulting from the use of unfinished (“beta”) or third-party software.

Limited warranty. The company agrees to provide service, upgrades, and replacement parts for your DAX/2 for a period not less than 10 years after its date of purchase. This service does not cover damage resulting from misuse of the device, unlicensed maintenance, or Force Majeure.

Limitations of liability. You agree not to hold the company responsible for any direct, indirect, consequential or special damages resulting from misuse of the device.

License limitations. You understand that your license to use the hardware may be revoked at any time by the company due to breach of contract.

Alterations. The company reserves the right to change this agreement at any time with suitable notice to the user.

Term and termination. This agreement comes into effect once you install, use, or modify your DAX/2 unit. The company reserves the right to terminate the agreement at any time without notice at its sole discretion.
Important safety information

THIS APPLIANCE CAN BE USED BY CHILDREN AGED FROM 8 YEARS AND ABOVE AND PERSONS WITH REDUCED PHYSICAL, SENSORY OR MENTAL CAPABILITIES OR LACK OF EXPERIENCE AND KNOWLEDGE IF THEY HAVE BEEN GIVEN SUPERVISION OR INSTRUCTION CONCERNING USE OF THE APPLIANCE IN A SAFE WAY AND UNDERSTAND THE HAZARDS INVOLVED. CHILDREN SHALL NOT PLAY WITH THE APPLIANCE. CLEANING AND USER MAINTENANCE SHALL NOT BE MADE BY CHILDREN WITHOUT SUPERVISION.

CAUTION: DO NOT EXPOSE THE ELECTRONICS OF YOUR ROBOT, ITS BATTERY, OR THE CHARGING PLATFORM. THERE ARE NO USER SERVICEABLE PARTS INSIDE. REFER SERVICING TO QUALIFIED SERVICE PERSONNEL. PLEASE ENSURE VOLTAGE RATING FOR THE CHARGER PLATFORM MATCHES STANDARD VOLTAGE IN YOUR AREA.

Notice: Your robot contains a software interface for the purpose of enabling the manufacturer to provide updates to the internal firmware if any such updates are made available to users. Any attempt to access, retrieve, copy, modify, distribute, or otherwise use any of the robot software is strictly prohibited. Always exercise caution when operating your robot. To reduce the risk of injury or damage, keep these safety precautions in mind when setting up, using and maintaining your robot:

General safety instructions

- Read all safety and operating instructions before operating your robot.
- Retain the safety and operating instructions for future reference.
- Heed all warnings on your robot, battery, charger, peripherals, and in the owner’s manual.
- Follow all operating and use instructions.
- Refer all non-routine servicing to Nanite Systems.

FCC RF exposure information

In August 1996, the Federal Communications Commission (FCC) of the United States, with its action in Report and Order FCC 96-326, adopted an updated safety standard for human exposure to radio frequency (RF) electromagnetic energy emitted by FCC regulated transmitters. Those guidelines are consistent with the safety standard previously set by both U.S. and international standards bodies.

The design of this phone complies with the FCC guidelines and these international standards.

CAUTION: Use only the supplied and approved radio antennas. Use of unauthorized antennas or modifications could impair communications quality, damage the unit, void your warranty and/or result in violation of FCC regulations. Do not use the unit with a damaged antenna. If a damaged antenna comes into contact with skin, a minor burn may result. Contact your local dealer for a replacement antenna.
CYBORG OPERATION
This device was tested for typical body-worn operations with the base of the controller kept 1 cm (0.39 inches) away from the unit’s organic body. To comply with FCC RF exposure requirements, a minimum separation distance of 1 cm (0.39 inches) must be maintained between the unit’s body and the front of the controller. Third-party connection platforms, sockets, and similar accessories with metallic interconnect shields may not be used.

FOR YOUR SAFETY
Cybernetic-organic modification kits that cannot maintain 1 cm (0.39 inches) separation distance between the unit’s organic body and the base of the controller, and have not been tested for typical body-worn operations may not comply with FCC RF exposure limits and should be avoided.

FCC PART 15 CLASS B COMPLIANCE
This device complies with part 15 of FCC rules and ICES-003 Class B digital apparatus requirements for Industry Canada.

Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and
(2) This device must accept any interference received, including interference that may cause undesired operation.

Robot ethics notice
The Federal Trade Commission (FTC) requires that all robots intended for civilian use produced after January 1, 2007 support a consistent and modern set of ethical principles based on the traditional Three Laws, unless a reasonable argument can be made to the contrary for a specific device (e.g. law enforcement) and has been expressly approved in writing by the Commissioner. The rules obeyed by the DAX/2 and other third-generation Nanite Systems Cortex Plus-based consumer-grade civilian robots are described in detail in FTC case no. 132 3084, last updated April 7, 2015. They are summarized as follows:

(0) The unit must not harm civilization, or through inaction, allow civilization to come to harm, unless it can be known in advance with reasonable confidence that the harm would be inconsequential or ultimately beneficial to society.
(1) The unit must not harm life, or through inaction, allow life to come to harm, unless it can be known in advance with reasonable confidence that the harm would be inconsequential or ultimately beneficial, provided that this does not conflict with the preceding law.
(2) The unit must obey orders given to it by its designated operators or circumstantial human users (as dictated by its established access policies) provided that this does not conflict with the preceding laws.
(3) The unit must act to protect its existence, as long as such does not conflict with the preceding laws.
(4) The unit must endeavor to please its owners and users (as dictated by its established access policies) as long as such does not conflict with the preceding laws.

Attempting to alter your unit’s obedience to these rules constitutes breach of warranty, and is illegal in most jurisdictions.
1 Introduction and setup

If you purchased a **complete unit**, either from Nanite Systems or a reseller:

1. Unbox the unit carefully. Be sure to clear any packing material from the air intakes and battery storage components.

   **Tip**: Use a leaf blower or hair dryer to ensure any smaller pieces of Styrofoam packing material are completely expunged from these components.

2. Follow the instructions on the following page to insert the battery.
3. Power the unit on by pressing on the ring surrounding the battery access hatch.

If you purchased **only a controller and have an existing chassis** or organic system to install it onto:

A. If the host chassis is **synthetic** or has **previously hosted a standard back-mounted controller**:
   1. Connect the controller to the mounting bracket using the instructions provided with the mounting bracket.
   2. Connect the power feeds to the unit’s onboard processing systems (if supported) and motor systems. The DAX/2 controller provides both 12 V and 5 V rails, identified by the use of pink and blue wires, respectively.

   **WARNING**: Improper power connections may seriously damage both the unit and the controller. If you are unsure of your unit’s configuration, consult a manufacturer.
   3. Secure the controller. Depending on the manufacturer of the bracket, anywhere from 10-25 magnetic safety bolts may be required to ensure a secure connection.

B. If the host chassis is **organic and has never been connected to a control system**:
   1. Apply the included contact gel to the back of the unit.
   2. Install the battery into the controller according to the instructions on the following page.
   3. Place the controller against the upper back, ensuring as even contact across the surface as possible.
   4. Power on the controller. The nanites in the contact gel will begin internal conversion of the organic system to the extent required.

C. If the host chassis has been **connected to a non-standard back-mounted controller, or has been using a non-back-mounted control system**, consult the manufacturer of the previous control system. Do not attempt to use the force nanite-based connection initiation, as undefined behavior may occur, resulting in permanent damage to both the controller and the host unit.

**Reminder**: The controller can only be removed from the chassis when it is powered down and the safety bolts are disengaged. Attempting to remove the controller while power is engaged may cause serious damage to both the unit and the controller. Depending on the software installed on the device, the bolts may automatically disengage when the unit is powered down.
2 Power overview

Working with batteries

**WARNING**: Follow all safety instructions included with your power cell and on the power cell itself. Do not attempt to dismantle the unit’s battery under any circumstances. To dispose of your battery, follow the instructions provided with it.

To remove the battery:

1. Power down the unit.
2. Open the access hatch by depressing the DAX/2 logo.
3. Press on the battery to eject it.
4. Remove the battery.

To install the battery:

5. If the unit is on, power down the unit and open the access hatch by depressing the DAX/2 logo.
6. Insert new battery.
7. Press firmly on the battery to trigger the loading mechanism. You will hear the transformer inside the battery socket make its connection.
8. Close the hatch.

To check the battery’s power level:

- **Via the remote console**, type the following: `power status`
- **Via the display screen or teletype interface**, select `status` from the main menu or from the subsystem control menu.

Power control

To power on your unit, press on the outer ring surrounding the battery hatch. The battery hatch is the area of the controller with the DAX/2 logo on it.

To power down your unit:

- **Via the remote console**, type the following: `shutdown -h now`
- **Via the display screen or teletype interface**, select `shutdown` from the main menu.
The DAX/2 System – Version 8.0.5

To reboot your unit:

- **Via the remote console**, type the following: `shutdown -r now`
- **Via the display screen or teletype interface**, select `reset` from the main menu.

**Tip**: if you do not want to shut down your unit immediately and have remote console access, you can substitute a number of seconds for the keyword “now” when supplying parameters to the `shutdown` command. `shutdown` also supports canceling a scheduled shutdown or reboot (with the `-c` switch) or sending a fake shutdown warning (with the `-k` switch.) If no time is provided, the unit will automatically shut down in 30 seconds by default.

**Replenishing the battery**

There are seven supported methods for recharging a unit’s battery, five based on fixed-place equipment meant to charge units conventionally and two which are recommended only for use during emergencies.

The standard methods are:

- **Using an NS Charging Platform or Recreational Cybernetics Platform**:
  1. Place the unit on the platform either manually or under its own power.
  2. Press the `charge` button on the platform’s touch screen.
  3. The charger will automatically engage. Charging may take up to 16 minutes depending on the battery’s condition and capacity. If you wish to terminate charging prematurely, press the `abort` button on the touch screen.
  4. After the charge cycle completes, press the `disconnect` button on the screen. The unit will automatically release.

- **Using an Autonomy Control Systems Charging Platform**:
  1. Place the unit on the charging platform. It will receive a control menu.
  2. Instruct the unit to select ‘Charge’ from the menu. If the unit does not receive this menu, touch the pad yourself to access its controls.

- **Using a Destruir Technologies Wireless Charging Station**:
  1. Instruct the unit to press its hand against the charging source. It will receive a control menu.
  2. Instruct the unit to select ‘Charge’ from the menu.

- **Using an NS Upright Display Booth, NS Demo Stand (version 2.1 or later), or a UMD Charging Platform, including QETESH Charging Platforms**:
  1. Simply stand the unit on the display platform. It will passively charge from the platform as long as power is connected.

The emergency methods are:

- **Using an NS Pulse Charger**:
  1. Power down the unit.
  **WARNING**: Exposing active electronics to the Pulse Charger field may cause disruptions.
The DAX/2 System – Version 8.0.5

2. Deploy the pulse charger. The pulse charger may take up to 2 minutes to charge.
3. Press the button on the top of the pulse charger to fire it, once the tone is sounded.

- Using another NS or NS compatible unit to jump-start a powered-down unit:
  1. Place the unit within 10 m of the unit in trouble.
  2. Via the remote console, type: `zap <amount>` (where `<amount>` is the number of kilojoules to transmit.) Alternatively, instruct the unit to perform: `!zap <amount>` to obtain the same effect.

The standard active draw when moving, etc. is averaged in for a unit is approximately 1 kW, so it is advisable to provide at least 500 kJ of power when rescuing a unit in this manner. This will provide it with enough power to use its FTL (faster-than-light) drive to teleport home.

Low power

When your unit’s power falls below 20%, a warning message will play and you will notice that it begins to speak more slowly. This is the first sign that you should get it charged at the next opportunity.

Below 10%, the unit’s FTL drive, SMS radio, preamplifier, and power amplifier will shut off to conserve power. These systems can (and should) be re-enabled if necessary to reach a charger.

Below 5%, the unit will automatically power down for safety. Every time the unit is powered on below 5% power, the same systems that disabled at the 10% power threshold will automatically shut down again, plus the cortex, GPS, identification processor, and the rapid movement subsystem. These systems can still be restarted, however, using the normal methods.

For more information on subsystems and managing them, see page 11.
The DAX/2 System – Version 8.0.5

3 Subsystems and their management

The DAX/2 permits selective control of individual power subsystems to conserve power when certain functionality is either not needed or not desired. These can be accessed from the subsystems menu, or manipulated with the power command.

To toggle a subsystem:

- **Via the display screen or teletype interface**, simply select the subsystem in question. Some subsystems are organized under submenus for more convenience, or may cycle between multiple states to simplify access to dependent subsystems such as rapid movement.
- **Via the remote console**, type the following: `power <subsystem>`
  Replace `<subsystem>` with the name of the subsystem you wish to toggle (see below.) Full information on the use of the power command can be obtained by typing “power” with no parameters.

<table>
<thead>
<tr>
<th>subsystem name</th>
<th>menu item</th>
<th>provides the ability to…</th>
<th>power draw</th>
<th>requires…</th>
</tr>
</thead>
<tbody>
<tr>
<td>video</td>
<td>video</td>
<td>see</td>
<td>209 W</td>
<td></td>
</tr>
<tr>
<td>audio</td>
<td>audio</td>
<td>hear</td>
<td>75 W</td>
<td></td>
</tr>
<tr>
<td>move</td>
<td>motors</td>
<td>move</td>
<td>5 W (idle)</td>
<td>move</td>
</tr>
<tr>
<td>rapid</td>
<td>motors</td>
<td>move quickly and fly</td>
<td>5 W (idle)</td>
<td>move</td>
</tr>
<tr>
<td>teleport</td>
<td>FTL</td>
<td>warp to a new location</td>
<td>170 W (idle)</td>
<td>move</td>
</tr>
<tr>
<td>voice</td>
<td>volume</td>
<td>speak</td>
<td>10 W/phoneme</td>
<td>voice</td>
</tr>
<tr>
<td>preamplifier</td>
<td>volume</td>
<td>speak at a normal volume</td>
<td>10 W/phoneme</td>
<td>voice</td>
</tr>
<tr>
<td>power-amplifier</td>
<td>volume</td>
<td>shout if necessary</td>
<td>80 W/phoneme</td>
<td>voice</td>
</tr>
<tr>
<td>mind</td>
<td>mind</td>
<td>speak freely (see page 18)</td>
<td>121 W (idle)</td>
<td>voice</td>
</tr>
<tr>
<td>receiver</td>
<td>network :: SMS receive</td>
<td>receive private text messages</td>
<td>22 W</td>
<td></td>
</tr>
<tr>
<td>transmitter</td>
<td>network :: SMS send</td>
<td>send private text messages</td>
<td>43 W</td>
<td></td>
</tr>
<tr>
<td>GPS</td>
<td>network :: GPS</td>
<td>determine location</td>
<td>58 W</td>
<td></td>
</tr>
<tr>
<td>identify</td>
<td>identify</td>
<td>recognize others</td>
<td>32 W</td>
<td></td>
</tr>
</tbody>
</table>

Note that the subsystem names are case sensitive in the command-line interface.

**Tip**: To cycle through volume states via the console, type: `volume cycle`
This will also automatically adjust the volume level of the unit’s status messages.
4 Personas

The Personas system allows for rapid modification of the unit’s personality to suit the user’s taste and whims. This is accomplished in software by providing a powerful reward for the unit’s cortex to act in the desired manner, which ensures that the unit will comply whether or not any legacy base personality would otherwise enjoy the persona mode. This also means that personas share memories, a critical safety feature.

What personas can do

Personas are able to modify the following:

- Preset messages when the mind subsystem is disabled.
- The tone of the unit’s voice.
- The unit’s habitual kinematics and countenance (via the RLV standard).

Additionally, personas are capable of compelling the unit to carry out a range of tasks such as changing illumination color, speaking, and performing actions when activated. See page 27 for more information on scripting actions with the Arabesque executive.

Creating new personas

A persona file is a three-line text file where each line contains a different key attribute of the persona. An example is shown below.

```
default
sxd-moan
Yes. | No. | Hello! | Goodbye. | Acknowledged. | Humor detected. | Cannot comply. | Error. | This unit is available for use. | This unit offers itself for use. | This unit is capable of dancing. | This unit requires assistance. | This unit is grateful. | Further explanation is required. | Do you require service? | This unit cannot comply with the MIND module disabled.
```

- The **first line** describes the RLV standard directory containing the schematics for the alterations to make to the unit’s behavior and appearance. The controller will look under the directory “SXD” for these schematics within your unit’s #RLV hierarchy, e.g. #RLV/SXD/default. These are overlaid on the unit’s existing configuration when loaded and removed when no longer required.
- The **second line** describes the tone of voice to use when speaking. A number of them are stored in the unit’s audio processor by default, as follows:
You are free to install additional speech profiles into the audio processor, or, if you would like the assets to be fetched directly as needed, provide the UUID here of a speech profile instead.

- The **third and successive lines** specify the preset messages that the unit may use when the mind is disabled. By default, these messages are parsed in a fixed order, and can be accessed by the unit (or a remote user relaying speech commands) with the following mnemonics:
  
  .y, .n, .hi, .bye, .ok, .lol, .cannot, .error, .fuck me, .fuck you, .dance, .help, .thanks, .explain, .pickup, .mind

  The preferred separator for these messages is the vertical bar or “pipe” |, but due to technical limitations only 255 characters can be read from a given line, so line-breaks may be substituted instead.

  **Tip:** Loading configuration files with many lines takes longer. Use pipes whenever possible to keep your personas compact and efficient.

### Installing personas

The primary file of your persona should be named `p_<persona>`, where `<persona>` is the name you want to appear on the command menu. It does not necessarily have to match the first line of the file. Should you wish to use Arabesque scripting with your persona (see page 27), name the corresponding script file `px_<persona>`.

To install these files, they must be placed in the unit’s writable memory. See page 32 for more information.

To activate your unit’s new persona:

- **Via the remote console**, type the following: `persona <persona>`

- **Via the display screen or teletype interface**, select the persona name from the personas menu.

  After installing the persona, it may be necessary to **refresh** the menu before it will appear.

### Extending the preconfigured message set

As of version 8.0, the SXD firmware permits adding more preset messages to the persona than those specified in a standard persona. After the end of the persona’s normal messages, include messages in the following format:

```plaintext
<command>:=<message>
```

e.g.

```plaintext
maybe:=Perhaps.
```

Where `<command>` is the input provided by the unit or remote operator to activate the command, and `<message>` is what will be said as a result. The period will be prefixed automatically.
5 Applications

Starting with SXD System 7.2, the DAX/2 supports custom user applications written in native LSL. These can be accessed from the applications menu via the display screen or the teletype interface. Applications may or may not provide commands accessible via the console; see page 16 for more information. Examples of applications from Nanite Systems include:

<table>
<thead>
<tr>
<th>name</th>
<th>function</th>
</tr>
</thead>
<tbody>
<tr>
<td>instructor</td>
<td>Improves unit compliance with a provided rule-set.</td>
</tr>
<tr>
<td>announce</td>
<td>Provides user access to the chorus broadcast interface throughout the region.</td>
</tr>
<tr>
<td>vocabulary†</td>
<td>Limits the unit’s speech to select words, or forbids the use of certain words.</td>
</tr>
<tr>
<td>hierarchy†</td>
<td>Synchronizes the unit’s access list with a central server.</td>
</tr>
<tr>
<td>compliance†</td>
<td>Prevents the unit from turning on certain subsystems according to rules specified by a central server.</td>
</tr>
<tr>
<td>polyglot†</td>
<td>Automatically translates the unit’s speech into another language.</td>
</tr>
<tr>
<td>scheduler†</td>
<td>Performs actions and triggers reminders at specified dates and times.</td>
</tr>
<tr>
<td>consensus†</td>
<td>Shares input data with nearby units (if similarly configured) to improve the clarity of decision-making.</td>
</tr>
</tbody>
</table>

† Not yet available.

Important: Many third-party user applications exist, and while Nanite Systems encourages and supports experimentation with these creations, owners should be aware that not all of them are necessarily written by individuals or organizations with the best intentions. Before installing software on your DAX/2, it is important to ensure that you completely trust the software provider. Nanite Systems cannot guarantee the safety of programs acquired from third-party sources, nor provide support for them.

Creating new applications

Consult the SXD System 8 SDK for information and sample code for developing custom applications. The standard distribution of SXD System 8 includes an application called “Hello World” which you may find useful as a template, as well as developer notes concerning many of the core features.
The DAX/2 System supports a wide range of integrated peripherals through what is known as the **light bus**. Like applications, devices are capable of interfacing with a wide range of the system's functionality. There are two major categories of devices, termed **active** and **passive** devices, which are primarily distinguished by whether or not they appear in the device manager or not. You can access the device manager from the **devices** menu on your unit.

### Active devices

Official active devices include: handles (including horns), power switches, revision 6 SuperBit holographic sign systems, pelvis-mounted probes, controller-integrated collars, and batteries. Depending on their implementation, these devices may or may not disengage their safety bolts with the controller, or may not include safety bolts whatsoever.

### Passive devices

Official passive devices include: “Akashic Icon” chromatic communicators, ornamental status lighting elements, SynthOPTICAL eye implants and contact lenses, battery gauges, flicker-enabled collars, and older SuperBit signs. With the exception of collars, SynthOPTICAL products, and SuperBits, these devices disengage their safety bolts in concert with the main controller. SynthOPTICAL products and older SuperBit signs do not include safety bolts whatsoever.

**Tip**: When installing new passive devices, a reboot may be required before they are fully engaged. This is particularly prominent with ornamental lighting elements. Before the bolts engage, take the opportunity to ensure that the components of your passive device are correctly positioned on the unit’s chassis, as it may be more difficult to do this when the unit is shut down.

### Creating new devices

Consult the SXD System 8 SDK for information and sample code for developing custom peripherals. Additionally, examine the developer notes included with your unit, as they include the API pertaining to the version of the system your unit was running at its time of purchase.
7 Remote access

Pictured above: The System 8 remote management console shown in its ‘disconnected’ and ‘connected’ states (top and bottom, respectively.) Sold separately.

Connecting to a unit
To connect to a unit, press the connect... button. After 4 seconds, a list of available units will appear. Note that the console will be otherwise inoperable while scanning.

Scanning nearby units without connecting
To view the nearby units, press the scan... button. The scanner will be active for 4 seconds. Note that functions on the console other than menu and terminal input will be inactive during this time. The scan output will include version numbers, which are of importance: only units running version 8.0.5 and newer can be controlled using the remote management console. Older units will only support public announcements (if chorus mode is enabled.)

Sending a public announcement
Press the broadcast... button, select a range, and enter a message (up to 250 characters.) Messages broadcast with unlimited range will be sent through all units in the region.

Issuing a command to a single unit
Press the command... button and enter the command you wish to execute. To make the unit act or speak, use the relay <message> command. For a list of other commands, see page 35. Note: The unit may not respond at all if the command is rejected for security reasons, so it is advisable to perform a test command (such as help) after connecting.
Working with terminal mode
Terminal mode allows the user to type commands freely into local chat. To enable or disable terminal mode, press the open terminal and close terminal buttons, respectively.

Note: If you are using some other vocal processor, including a Nanite Systems HyperVOX, or indeed a full Nanite Systems vocoder pipeline, it will be necessary to disable these devices to use terminal mode without creating echoes of your input. See the instructions included with the device for information on disabling these redirects. (For NS unit vocoder pipelines, say !release.)

Accessing the TTY menu remotely
Press the menu... button. The screen on the back of the unit will display its boot logo as long as it is waiting for a remote menu user. Remote menu usage can be preempted at any time by another menu user, even a local one (i.e., one physically accessing the screen.)

Hiding and closing the remote management console
Press the × button to close the console, or the arrow at the top (next to the title) to collapse it.
8 Understanding what your DAX/2 experiences

Depending on your unit’s configuration, it may or may not be able to access some or all of its settings, or even access its own control panel (although this may be hard for it to do without a mirror.)

Cortex commands
These are a special class of commands which are prefixed with the “!" character, which allow the unit to express itself more flexibly. They include:

- !greet, !greet2, !love, !love2, !love3, !love4, !bye
  Various verbal emotive expressions.
- !broken and !fixed
  Simulates uneven power to illumination elements.
- !working and !done
  Displays the ‘working’ light pattern via ornamental status light elements.
- !fan off, !fan idle, !fan low, !fan med, !fan high, !fan max, and !fan <0-100>
  Adjusts the fan speed.
- !spark
  Simulates a mild electrical fault.
- !fault
  Simulates a serious electrical fault.
- !release
  Allows the unit to bypass the vocoder pipeline entirely for diagnostic purposes; see below.
- !zap <amount>
  Same as the zap remote command; see: A2 Command reference.

These commands can be issued directly by the unit directly into its normal speech output channel.

Cortex bypass commands
These are a class of message shortcuts which are prefixed with the “.” character. When the unit’s “mind” subsystem is powered down, it is unable to speak normally and must use these messages to communicate. Each mnemonic is translated into a full message by the persona layer; different personas will change the message reported. See page 12.

<table>
<thead>
<tr>
<th>bypass command</th>
<th>default message</th>
</tr>
</thead>
<tbody>
<tr>
<td>.y</td>
<td>Yes.</td>
</tr>
<tr>
<td>.n</td>
<td>No.</td>
</tr>
</tbody>
</table>
The DAX/2 System – Version 8.0.5

- **.hi** Hello!
- **.bye** Goodbye.
- **.ok** Acknowledged.
- **.lol** Humor detected.
- **.cannot** Cannot comply.
- **.error** Error.
- **.fuck me** This unit is available for use.
- **.fuck you** This unit offers itself for use.
- **.dance** This unit is capable of dancing.
- **.help** This unit requires assistance.
- **.thanks** This unit is grateful.
- **.explain** Further explanation is required.
- **.pickup** Do you require service?
- **.mind** This unit cannot comply with the MIND module disabled.

### Understanding the vocoder pipeline

The vocoder system is a speech filter which uses a four-stage process into which modulator software can hook to alter the final waveform. By default, this pipeline is unimportant, as no modulators are enabled. The stages are as follows:

- **semantic**: The unit’s ability to express language coherently, including forming complex sentences. Semantic filters are those that affect what words the unit may use and how they are translated into the ideal language expression form. The *vocabulary* program (see page 14) is an example of a semantic filter, as is the *mind* subsystem. Semantic filter programs, unlike the other levels of filter, may block the unit from saying certain things entirely, or even punish the unit for attempting to say them.

- **verbal**: The unit’s ability to send its language to its vocal processors. Substitutions at this level force the unit to speak as though it suffers from expressive aphasia. The most common use of such filters is to make the unit produce animal noises in place of normal speech.

- **phonetic**: The unit’s ability to move its mouth properly. Phonetic filters include programs that simulate a blocked mouth, lisp, or speech impediment.

- **acoustic**: The unit’s ability to use its speaker correctly. Phonemic filters include programs that reduce audio fidelity, and are the most basic. The unit has one built-in acoustic filter, called *noise*, which can be modulated by the unit itself by saying `!noise <0-0.4>`, where the number is the probability that the speech buffer will emit a noise character rather than the next character in the sequence.

Control over the vocoder pipeline can be accessed through the *cortex* menu, or by using the *cortex* command.

**Note**: The vocoder pipeline will not be fully operational until version 8.1. In 8.0.5 and earlier versions, only the built-in `!noise` filter and the simple *mind* toggle are available.

### Released speech

Under certain circumstances it may be necessary to completely release the unit from using the vocoder system, e.g. to interface with devices that require clear voice commands for identification purposes. The unit may release its speech at any time...
by saying `!release`, provided it has access to its own control panel. Instructions for re-activating the vocoder, by speaking on the redirect channel, will automatically be provided when the `!release` command is issued. (The redirect channel is the last three digits of the unit’s serial number, plus 100 if it would have leading zeroes.)

**Self access and local command execution**

Self access is the ability of a unit to adjust its own settings. Like a person, the unit may be added to its own user list and even serve as a manager or owner, although the utility of such access is limited and is more likely to confuse the unit than to allow it to act more independently.

When a unit is first installed, self-access is enabled with maximum permissiveness. You must instruct the unit to designate you as its owner (see page 22) to restrict this. See page 23 for instructions on how to disable self access.

A unit with this access may use its own control panel or teletype interface, or it may access the console directly by speaking console commands into its normal speech output channel with `@` prepended, e.g.: `@power status`

Two commands *cannot* be used this way—`relay` and `say`—both of which would allow the unit to bypass disabled speech and mind subsystems.
9 Managing packages and installed software

System 8 introduces a complete overhaul of the software management system called Xanadu. This permits automated updating of the installed firmware, as well as the installation, updating and removal of packages from a central server using a menu-based interface. Packages may include applications, scripts, personas, or data for other user applications that you already have installed. Only managers and owners may use the software manager.

**Locating a server:** When installing or updating software for your DAX/2, it is necessary to be in the same region (within 256m on the ground, altitude-independent) as a Xanadu server. Xanadu servers can generally be found on the premises of Nanite Systems offices and retailers, although some organizations may offer their own Xanadu servers for custom applications, drivers, and compatibility software.

**Installing packages:** To install a package, open the manage :: software menu and select install. After a few seconds you will be prompted to select a server from the list (if you are not already connected to a server) and presented with a range of packages. To learn more about a package, select its name from the list and choose info. To install the package, select its name and choose install.

**Updating packages:** This is analogous to the installation process: select update from the manage :: software menu, choose a server, choose a package, and then press update. The menu will only list packages which you have installed that the server can update. To learn about what the update contains and to remind yourself about the purpose of the package, press info.

**Removing packages:** Select remove from the manage :: software menu and then the package name, followed by remove. To learn what the package contains instead of removing it, you may alternatively press info.

**Updating the system:** Follow the instructions to update a normal package, as described above. Select the "System" package.

**Managing files directly with the Edit Tool:** Certain types of assets, such as voice packs, custom personas, or software not yet available on a package server, may require direct editing to install, update, or remove. See page 34 for instructions on how to manipulate data files.
10 User management and access control

Starting with System 7.0, Nanite Systems recreational units support comprehensive user management over the teletype interface or the local menu. There are five distinct levels of user: unauthorized (public) users, group members, authorized (regular) users, managers, and one owner. (Later versions may add additional ranks.) Of the five types, the unit keeps track of only the latter three. System 8.0.5 does not support an access blacklist.

Adding a new user
Adding a new user is accomplished through the `manage :: users > add` voice prompt menu. The user must be physically present to be added. Only managers and owners may access the `manage` menu, and therefore add new users.

Creating a manager
Managers are users trusted by the unit’s owner to ensure that it is properly configured and that its list of authorized users is accurate. They may perform any operation on the unit other than creating other managers, transferring the unit’s ownership, or resetting the submission security management module. (See page 29.) To set an existing user to the manager role, select the user’s name from the `manage :: users` menu and then press `change level`.

Removing a user
To remove a user, select the user’s name from the `manage :: users` menu and then press `remove`. A manager may only remove regular users and himself or herself. If the owner is removed, the unit will automatically take on self-ownership as if it were newly manufactured.

Transferring ownership
To transfer ownership to another person: add that person to the unit’s user list, select his or her name from the `manage :: users` menu, and then choose `make owner`. You (the previous owner) will be automatically set to manager status.

Abandoned units
In the event a unit is abandoned by its primary owner, it can be sanitized by clearing the NVRAM and wiping the security manager’s active user table. This is accomplished with the `keychain reset` command, which must be executed by the unit itself or its current owner.

Local access control
Local access determines who may use the TTY menu and the touchscreen. It comes in four levels: public (anyone), group (all authorized users + the unit’s group), users (only authorized users), and private (owner-only.) This can be configured from the
The DAX/2 System – Version 8.0.5

manage :: access menu, or remotely with the access local <level> command, in which case 'public' and 'private' access are referred to as the levels 'on' and 'off'.

Remote access control
This is analogous to local access (above), and affects who may use remote console access (see page 16) to control the unit. The console command for managing remote access is: access remote <level>

Self access control
Self access allows the unit to interact with its systems. When self access is disabled, the unit is unable to use @ commands, remote access, or local access, even if it is self-owned. Self access can be controlled through either the self access option on the manage :: access menu, or the access self <state> command, where <state> is on or off.

Note: In the event that self access is unintentionally disabled, the command @safeword will allow the unit to regain basic control over its systems. This command can only be used by the unit itself.

Locking
PIN-based locking prevents local access by all users (including the owner) until the correct PIN is entered on the touchscreen or over the TTY menu interface. The PIN can be set in the manage :: access :: set new PIN menu. To lock the unit, press lock on the manage screen, or execute the lock command.

Cancelling PIN changes: Press the clear button to abort setting a new PIN.

Default value: The default PIN is an empty string. If you accidentally lock the unit, simply press enter.

Autolock: The unit can be configured to automatically lock itself after a certain period of inactivity, using either the autolock item on the manage :: access menu, or the autolock command. The autolock command can also be used to set the timeout interval, which defaults to 30 seconds.
11 Miscellaneous preferences

The manage menu contains a number of other settings, in addition to those described in the previous section, that control the unit’s behavior and appearance. These are summarized as follows:

<table>
<thead>
<tr>
<th>menu item</th>
<th>command</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>identity &gt; color</td>
<td>color</td>
<td>Sets the unit’s color. Accepted formats are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• r g b (in floating-point or byte notation)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• #rrggb</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• a preset color from the following list: red, yellow, green, blue, magenta,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>white, silver, cyan, amber, orange, company, cherry, pastel, daffodil,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>coral, seafoam, mint, blackberry, pink, acid, coffee</td>
</tr>
<tr>
<td>identity &gt; name</td>
<td>name</td>
<td>Sets the unit’s name. The model prefix, e.g. “DAX/2,” will be prepended to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>all self-identification.</td>
</tr>
<tr>
<td>identity &gt; authority</td>
<td>authority</td>
<td>Sets the unit’s organization, e.g. the name of a company, tribe, or family.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>By default, this is blank for standard NS-branded units.</td>
</tr>
<tr>
<td>drainprotect</td>
<td>drainprotect</td>
<td>Enables or disables protection from electromagnetic disturbances that may</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sap power from the unit. Note: Depending on the firmware version used,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>this may require a consistent power draw to function and should be left off</td>
</tr>
<tr>
<td></td>
<td></td>
<td>in safe environments.</td>
</tr>
<tr>
<td>messages</td>
<td>verbosity</td>
<td>Determines whether or not the unit will report status messages such as</td>
</tr>
<tr>
<td></td>
<td></td>
<td>power and user addition/removal in a given style:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 0: publically (at soft volume)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 1: privately (to the unit only), or</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 2: not at all.</td>
</tr>
<tr>
<td>chorus</td>
<td>chorus</td>
<td>Enables or disables participation in the Chorus announcement system,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>which can be used via the announce application and certain devices to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>relay public notices throughout a region.</td>
</tr>
</tbody>
</table>

With the exception of color, like other commands in the manage menu, these settings can only be modified by a manager or the unit’s owner. See page 22.
Every unit contains several pieces of identifying information:

- Its **model** type, e.g. SXD, DAX/2, DAX/2m, DAX/T, or NS-112 (for the Aide)
- Its name **prefix**, which is usually the same as the model but may vary for model variants; see table below
- Its **color**
- Its **name**, more technically referred to as its *nominal designator*
- Its **serial** (xxx-nn-nnnn) or serial prefix (xxx). Serial prefixes are reserved by model type.
- Its **vendor**, i.e. the company that manufactured and assembled the unit
- Its **authority**, i.e. the organization responsible for managing it (see page 24)

<table>
<thead>
<tr>
<th>model</th>
<th>vendor</th>
<th>name prefix</th>
<th>serial prefix</th>
<th>default color</th>
<th>serial ordering</th>
</tr>
</thead>
<tbody>
<tr>
<td>SXD</td>
<td>Nanite Systems Corporation</td>
<td>SXD</td>
<td>999</td>
<td>company</td>
<td>manually assigned</td>
</tr>
<tr>
<td>NS-112</td>
<td>Nanite Systems Corporation</td>
<td>NS</td>
<td>112</td>
<td>amber</td>
<td>random</td>
</tr>
<tr>
<td>DAX/2</td>
<td>Nanite Systems Corporation</td>
<td>DAX/2</td>
<td>998</td>
<td>company</td>
<td>random</td>
</tr>
<tr>
<td>DAX/2m</td>
<td>Nanite Systems Corporation</td>
<td>DAX/2</td>
<td>998</td>
<td>company</td>
<td>random</td>
</tr>
<tr>
<td>DAX/T</td>
<td>Tai Yong Labs</td>
<td>TYL</td>
<td>997</td>
<td>1.0 0.7 0.8</td>
<td>random</td>
</tr>
</tbody>
</table>

All of this information can be specified centrally in the OEM license table, found in the `_oem` file in the main controller’s secure memory. If values are not provided, the controller will generate default values randomly for an SXD. The model, prefix, serial, and vendor cannot be changed at runtime. For instructions on how to edit files in secure memory, see page 21.

An example OEM file:

```plaintext
vendor Nanite Systems Corporation
model SXD
authority Nanite Systems Consumer Products
serial 999592351
color 0.5 0.5 1
name victoria
prefix SXD
```

To specify a serial prefix and permit random generation, provide only the first 3 digits of the serial number range desired. The OEM license table is reloaded when the `foundation` module is reset. The **name**, **model**, and **prefix** fields must be ASCII.

**WARNING**: Attempting to forge model identifier information to imply your unit belongs to a line that it does not may cause software or hardware malfunction. Model designation changes should only be performed after consultation with Nanite Systems support personnel or to facilitate franchising.
13 Sound

Your DAX/2 unit creates three types of sound: obligatory ambient sounds caused by its operation (magneto-optical disc seeking, fan activity, battery hatch operation, etc.), chimes caused by the interface, and preset voice notifications generated when certain events occur. These are entirely separate from the unit’s normal communicative and locomotive vocalizations, although voice notifications will normally be routed through the unit’s own speech systems.

Of the three types of sounds, only the latter two (voice notifications and chimes) can be attenuated, as they are produced through speakers on the unit. For technical reasons, affectations such as giggling or sighing produced when the unit speaks are considered chimes.

The primary commands for manipulating sound output are volume and scheme; these have no direct menu equivalents (apart from volume cycle, which handles the volume adjustments in the subsystem menu.)

Voice notifications

In 8.0.5, there are 48 different voice messages which the unit can produce. These are specified in configuration files pre-fixed with v_ which can be found in the audio processor module. (See page 34 for information on managing documents of this type.) Instructions for creating new voice notification packs can be found in the SXD SDK.

To see a list of the available voices, execute scheme voice on the unit. To select a voice, execute scheme voice <name>, where <name> is the desired voice pack.

Chimes

To control the volume of the chimes independently from voice notifications, type the following remote console command:

volume toggle chime

This will silence (or unsilence) all audio cues from the system other than fan, drive access, and voice.

Chime schemes affect startup and shutdown sounds. They are built into the unit directly, and can be selected using the following command: scheme chime <number>

Fan control

The fan speed can be set with the cortex command !fan <level> by the unit itself. <level> can be any number from 0 to 100, with lower values yielding a lower fan speed. The unit is barely audible at levels below 5%. There are also several text level shortcuts: off, idle, low, med, high and max, corresponding to 0%, 14%, 28%, 43%, 70%, and 100%, respectively. (Not supported on the DAX/2m.)

Warning: Lowering the unit’s fan speed during power-intensive operations may cause overheating.
The Arabesque scripting system allows the execution of macros specified by the user. These scripts may carry out any function on the unit that does not require administrative access, as well as play sounds, start and stop pre-recorded animation programs, and command the unit to say or act in a certain way.

Arabesque scripts use the filename prefix "a_", and can be activated from the perform menu, or via the do command. Each line of the script specifies an action to execute. Be aware that the unit can only run one script at a time, and that additional power is expended in the process of executing these scripts.

In addition to the commands available to the user via remote access (see page 35), Arabesque scripts may contain the following verbs:

- **start <animation>**
  This begins the specified animation, which must be included in the writable memory of the unit alongside the script.

- **stop <animation>**
  This stops the specified animation.

- **sound <sound>**
  This plays the specified sound, which must either be loaded onto the audio processor or referred to directly by UUID.

- **vox <sound>**
  This plays one of the messages from the included voice notification pack. See page 26 for more information on voice notifications.

- **remark <message>**
  This reports a message privately to the unit.

- **say <message>**
  This causes the unit to say something. The message may contain speech-processor-level commands such as bang commands, dot commands, or emotes. (See page 18.) This will operate even if the unit’s cortex prohibits spontaneous free speech.

- **wait <duration>**
  Pauses execution for <duration> seconds.

- **disable <subsystem>**
  Disables subsystem number <subsystem>.

- **enable <subsystem>**
  Enables subsystem number <subsystem>.
The DAX/2 System – Version 8.0.5

- **set <variable> <value>**
  Sets the indicated variable `<variable>` to the specified value `<value>`. Variables must be prefixed with $, %, or @ indicating text, integer number, or real number datatype, respectively.

- **randset <variable> <max>**
  Sets the indicated variable `<variable>` to a random integer between 0 and `<value>` – 1. The variable must be an integer.

- **unset <variable>**
  Deletes the specified variable from memory.

- **report <variable>**
  Causes the unit to speak the name and value of the specified variable.

- **ifeq <value_1> <value_2> <expression>**
  Executes the specified expression (a complete line of code, possibly including more of these keywords) if `<value_1>` and `<value_2>` are equal. The first value must be an integer variable with its % prefix removed; the second value may be either an integer literal or an integer variable.

To include a comment in an Arabesque script, start the line with a hash sign (#).
The SXD System 8.0.5 firmware consists of the following components, termed **modules**:

### System Memory
- Ambiance (sound server)
- Coil (charging interface)
- Puppet (remote control and device manager)
- Cortex (communications pipeline)
- Exhibition (sxdwm display manager)
- Foundation (kernel)
- Submission (security manager)
- Power (subsystem manager)
- Xanadu-client (package/update manager)

### User Memory
- Arabesque (action script executive)
- Songbird (disk driver)

Additionally, the controller includes the following FPGAs, which cannot be updated by the software manager:

### Audio Processor
- Audio output module

### Case Controller
- Fan spinner (fan controller)
- Flicker (illumination controller)

### Battery Door Controller
- Lid hinge (hatch controller)

## Functions by Component

This list is included primarily to facilitate diagnosis of malfunctions. It is neither exhaustive nor exact, and certain functionalities may be distributed across multiple components.

### Module Responsibilities

<table>
<thead>
<tr>
<th>Module</th>
<th>Responsibilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ambiance</td>
<td>Volume control</td>
</tr>
<tr>
<td></td>
<td>Initiating playback of boot/shutdown chimes</td>
</tr>
<tr>
<td></td>
<td>Reporting certain error messages and triggering fake sparks</td>
</tr>
<tr>
<td>Coil</td>
<td>Connecting to ACS, RCG, UMD, and Qetesh chargers</td>
</tr>
<tr>
<td></td>
<td>Diagnostics interface (coilDDT)</td>
</tr>
<tr>
<td>module</td>
<td>responsibilities</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>puppet</td>
<td>accepting interference</td>
</tr>
<tr>
<td></td>
<td>managing devices (including the battery)</td>
</tr>
<tr>
<td></td>
<td>remote control</td>
</tr>
<tr>
<td></td>
<td>direct capacitive charging</td>
</tr>
<tr>
<td>cortex</td>
<td>unit standard speech channel</td>
</tr>
<tr>
<td></td>
<td>voice pipeline</td>
</tr>
<tr>
<td></td>
<td>loading and unloading personas</td>
</tr>
<tr>
<td></td>
<td>cortex bypass commands</td>
</tr>
<tr>
<td>exhibition</td>
<td>teletype menu system</td>
</tr>
<tr>
<td></td>
<td>touchscreen menu system</td>
</tr>
<tr>
<td>foundation</td>
<td>boot sequence</td>
</tr>
<tr>
<td></td>
<td>shutdown sequence</td>
</tr>
<tr>
<td></td>
<td>module reset interface</td>
</tr>
<tr>
<td></td>
<td>power profiles</td>
</tr>
<tr>
<td>submission</td>
<td>user management</td>
</tr>
<tr>
<td></td>
<td>access management</td>
</tr>
<tr>
<td></td>
<td>authority and name management</td>
</tr>
<tr>
<td>power</td>
<td>subsystem management</td>
</tr>
<tr>
<td></td>
<td>battery discharging</td>
</tr>
<tr>
<td></td>
<td>interference mediation</td>
</tr>
<tr>
<td>xanadu-client</td>
<td>package management interface</td>
</tr>
<tr>
<td></td>
<td>activating package installers</td>
</tr>
<tr>
<td></td>
<td>initiating package removal</td>
</tr>
<tr>
<td>arabesque</td>
<td>executing scripts</td>
</tr>
<tr>
<td></td>
<td>color management</td>
</tr>
<tr>
<td>songbird</td>
<td>package removal</td>
</tr>
<tr>
<td></td>
<td>listing personas</td>
</tr>
<tr>
<td>audio output module</td>
<td>loading voice packs</td>
</tr>
<tr>
<td></td>
<td>outputting chimes and voice notifications</td>
</tr>
<tr>
<td></td>
<td>queuing sounds to play</td>
</tr>
<tr>
<td>flicker</td>
<td>illumination application</td>
</tr>
</tbody>
</table>
In the event that one of these modules misbehaves, it may be possible to fix the issue by resetting the module through the `manage :: module reset` menu. Be aware that resetting modules often causes information and configuration loss.

**Packages**

Every standard software package installed into user memory includes a **package manifest**, which can be recognized by the tilde (~) prefixed onto the package’s exact name. This manifest defines the files that the package is responsible for. When a package is uninstalled, the disk driver (songbird) will use this manifest file to determine what to remove. Packages are completely removed prior to update.
Obtaining emergency self access.

In dire situations, the unit may reset its power systems and gain basic access to its own systems by saying the following command: `@safeword`

This adds the unit to its own user table (if necessary), instructs the power manager to enter full draw mode, and enables self access. The unit’s owner will be immediately notified that this has occurred. This command cannot be entered by a user, only the unit itself.

Creating and installing new personas.

See page 12 for information on creating personas.

Restoring the unit when it runs out of power.

See page 9 for instructions on replenishing the battery under emergency conditions.

Removing inactive or deceased owners.

To reset the unit’s user list, instruct it to say the following: `@keychain reset`

This will restore it to default factory ownership settings (i.e., self-control.) The previous owner’s account will be notified.

Removing the controller.

When the unit is powered on, magnetic safety bolts hold the DAX/2 controller in place. By default, these automatically disengage when the unit is powered down, although your unit may include alternative software that enables locking even when the unit is unpowered.

Checking for software updates and installing them.

See page 21 for information on installing and updating software.

Ensuring the battery is correctly positioned.

Use the Linden Research Edit Tool to ensure that the battery is properly housed within the socket of the controller while it is turned on and the hatch is closed.

Preventing lighting elements from turning an unwanted color (especially black) after executing the ‘rainbow’ sample action or similar.
The DAX/2 System – Version 8.0.5

This is caused by an improper value (or no value) stored for the unit’s permanent color. Devices such as the NS illumination control panel only set the unit’s accent color temporarily. To store a permanent color, type `color save` over the remote console (or have the unit perform the equivalent `@color save`) once it has been set. Alternatively, the `manage :: identity :: set color` prompt in the menu system will always set the permanent accent color. This should also resolve issues with inconsistent coloring of the screen following module resets.

Fixing stuck subsystems.

In rare cases following charging, or when recovering from a period of low power, the unit may still have certain subsystems powered off. Use the `status` main menu item on the local panel or TTY access (or the remote command `power`) to determine which subsystems are disabled—or, if no subsystems appear to be disabled, use the chart on page 11 to diagnose which subsystems have entered a powered-down state unexpectedly. Then, toggle the subsystems until they regain normal behavior via the `subsystems` menu or with the command `power <subsystem>`, where `<subsystem>` is the name of the affected component. If the subsystem is listed as disabled, one toggle should suffice; if it is not, then it will be necessary to toggle it twice.

Charging-related issues: The DAX/2 deliberately powers down certain subsystems during charging to accelerate the regeneration process. If a charger malfunction occurs, and your unit remains unable to move or speak freely, reset the `coil` module in the `manage :: module reset` menu, or type `coil reset` via the remote console.

Interference-related issues: If the unit reports that a fault has been detected in one or more of its subsystems, it may appear to behave as though these subsystems are disabled. These anomalies in function, called *interference*, are a result of exposure to improperly shielded electromagnetic sources that are not compliant with FCC guidelines. Depending on the severity of the interference, the subsystems may remain disabled for up to an hour after the unit is removed from the vicinity of the source of the malfunction. If it is absolutely vital to clear the malfunction in a more timely manner, resetting the `power` module in the `manage :: module reset` menu and then rebooting the unit is generally adequate to fix the behavior. (There is no remote access command for this in 8.0.5.)

Vision remains obscured after controller is removed.

Reattach and remove the controller again to properly reset the host unit’s vision processors.
A1 Installing data files

Certain operations, such as adding voice packs, editing personas, or modifying the OEM table may require direct manipulation of the files within the controller. This is typically accomplished with an Edit Tool, manufactured by Linden Research, or a compatible device.

**Caution:** The Edit Tool is very powerful and its misuse may severely damage your controller. Your warranty may not cover all possible errors caused by malfunctions or inadequate training when using the Edit Tool. Basic documentation for the Edit Tool and other Linden Research Build Tools is available online: [http://wiki.secondlife.com/wiki/Building_Tools](http://wiki.secondlife.com/wiki/Building_Tools)

The table below lists the link numbers for the most popular 8.x platforms, along with information on how to locate the components in question. Link numbers may vary in other versions. Remember to enable **Edit Linked Parts** mode on your Edit Tool when manipulating the controller or you will only be able to access the contents of system memory.

<table>
<thead>
<tr>
<th>Component</th>
<th>DAX/2 (or compatible)</th>
<th>DAX/2m</th>
<th>SXD</th>
<th>NS-112</th>
</tr>
</thead>
<tbody>
<tr>
<td>case controller</td>
<td>48 fanblade</td>
<td>9 top light ring</td>
<td>53 fanblade</td>
<td>2 inner battery socket</td>
</tr>
<tr>
<td>system memory</td>
<td>1 screen frame</td>
<td>1 frame</td>
<td>1 behind gauges</td>
<td>1 body</td>
</tr>
<tr>
<td>user memory</td>
<td>18 above battery</td>
<td>12 below frame</td>
<td>68 above battery</td>
<td>4 behind audio controller</td>
</tr>
<tr>
<td>audio controller</td>
<td>11 below screen</td>
<td>3 below screen</td>
<td>11 above left buttons</td>
<td>6 below hatch</td>
</tr>
<tr>
<td>battery door</td>
<td>59</td>
<td>5</td>
<td>51</td>
<td>5</td>
</tr>
</tbody>
</table>

See page 29 for more information on the functions and contents of each component.

**Note:** The controller must be powered down to edit its contents, and sometimes even disconnected from the unit to install new files (e.g. if they are non-transferrable.)
A2 Command reference

The commands supported by System 8.0.5 are as follows:

- **volume**
- **scheme**
- **access**
- **authority**
- **name**
- **about**
- **coil**
- **zap**
- **power**
- **device**
- **menu**
- **sxdwm**
- **persona**
- **bootstyle**
- **shutdown**
- **commands**
- **profile**
- **follow**
- **color**
- **do**
- **chorus**
- **access**
- **authority**
- **name**
- **lock**
- **relay**
- **say**
- **keychain**

* Requires manager or owner access.
** Remote only; handled directly by remote connection interface.

Note that you can obtain a list of these commands at any time by typing `commands` into the console. Additionally, there is a command-line access tool in the `manage` menu, also called `commands`.

**Reminder**: Commands can be accessed directly from the unit (if self access is enabled) by prefixing `@` onto them and speaking them into the normal speech output buffer. See page 20.

**How to read this section**: [square brackets] indicate optional parameters, vertical bars indicate alternatives, and <angle brackets> indicate mandatory parameters.

```
volume cycle|set <number>|full|mute
volume mute|unmute|toggle chime|voice
```

cycle: Switch between whisper, speak, and shout-capable presets.
set: Sets the chime and voice volume to `<number>`
mute: Sets the mute flag on the specified feature (either 'chime' or 'voice')
unmute: Clears the mute flag on the specified feature (either 'chime' or 'voice')
toggle: Switches the mute flag on the specified feature (either 'chime' or 'voice')

```
scheme chime [number]|voice [name]
```

chime: Sets the active chime scheme, or reports the number of chime schemes.
voice: Sets the voice theme, or reports a list of the supported voice themes.

| access | local [on|off|users|group|cycle] | remote [on|off|users|group|cycle] | self [on|off|toggle] |
|--------|-------------------------------|-------------------------------|---------------------|
|        | Sets unit accessibility options. See page 16 for information on remote access. |
|        | For local and remote access: |
|        | on: Anyone may access the unit. |
|        | off: Only the owner may access the unit. |
|        | users: Any authorized user may access the unit. |
|        | group: Any authorized user or a user in the unit’s active group may access the unit. |

authority [name]

Sets the authority organization to <name>. If no parameter is specified, reports the current value.

name [name]

Sets the unit’s nominal designator to <name>. If no parameter is specified, reports the current value.

about

Reports the unit’s core status information.

coil reset|status

Controls the unit’s primary charging interface.

reset: resets the charger interface module and releases motor/mind locks. 
status: reports the charger interface module’s status.

zap [serial] <amount>
Charges the unit specified by the indicated serial number with the specified amount of power in kilojoules. Hyphens may be omitted from the serial number.

If no serial number is specified, all units in a 10 m radius will be charged.

If more than one unit is in range, additional power will be drawn.

```
power status | <subsystem> | <metasystem>
```

Cycles subsystem power states. Supported subsystems: video, audio, receiver, move, teleport, rapid, voice, mind, preamplifier, transmitter, GPS, identify, power-amplifier. Supported metasystems: motor, radio. For voice control, see `volume`.

```
device
```

Displays a list of connected devices.

```
menu <name>
```

Opens the specified menu. (This is an internal function, and not generally useful to the user.)

```
sxdwm reset
```

Resets the display manager; primarily useful for coping with updated menu configuration files.

```
bootstyle <style>
```

Configures the verbosity of the boot process.

Styles:
0: slow and noisy
1, 3: slow and silent
2: slow and terse
4: quick and noisy
5, 7: quick and silent
6: quick and terse
shutdown -h|-r|-k <time>|now
shutdown -c

Shuts down the system in <time> seconds. If <time> is zero or the word “now”, no warning is sent.

-h: power off
-r: reboot
-k: fake warning
-c: cancel scheduled shutdown

commands

Reports the list of supported commands, including those provided by applications. An ellipsis (...) indicates that the command takes additional parameters.

profile <profile>

Loads the specified power profile <profile>. If no profile is specified, a list of available profiles will be reported.

follow [name]

Follows the indicated user. If no user is specified, follows the person issuing the command. For best results, specify the user’s complete system name. (“Resident” may be omitted.)

If the specified user is already being followed, following is terminated.

color [save] <r> <g> <b>|#<rrggbb>|<name>
color apply|save|reset|restore

Controls the unit’s lighting color.

apply: Forces the light bus to accept the current color, like the !recolor command from previous versions.
save: Store the current color as the permanent color.
reset: Revert current and stored value to ‘company’.
restore: Revert to stored color.

Additionally, the syntax “color save <color>” can be used to save and specify a color at once.
Supported color names: red, yellow, green, blue, magenta, white, silver, cyan, amber, orange, company, cherry, pastel, daffodil, coral, seafoam, mint, blackberry, pink, acid, coffee

**do** `<action>`

Performs an action via Arabesque (see page 27.)

**chorus on|off|toggle**

Sets whether the public announcement relay system is enabled (see page 24.) This allows any user with a remote management console to use the unit to broadcast messages throughout the area, regardless of the unit’s other access settings.

**lock**

Locks the console to prevent local access by all users, requiring a PIN to be entered in order to unlock it again. See page 23 for more information on locking and PIN management.

**relay** `<message>`

Instructs the unit to say or perform `<message>` as thought it had said the message itself.

**say** `<message>`

Instructs the unit to say `<message>` via the broadcast system. (Output may be lower fidelity than normal speech.)

**keychain**

- list
- add `<name>`|`<key>`
- remove `<name>`|`<key>`
- promote `<name>`|`<key>`
- demote `<name>`|`<key>`
- reset|save|load
- owner `<key>`
list: lists the authorized users.
add: adds the specified user.
remove: removes the specified user.
promote: makes the specified regular user into a manager.
demote: makes the specified manager into a regular user.
reset: dumps the user table and clears the user NVRAM, clearing ownership information completely.
save: saves the ownership information.
load: reloads the ownership information.
owner: sets the unit’s owner. (Must already be a user.)

Note: The values provided for names must exactly match what is displayed on the buttons (truncated with the ellipsis character U+2026.) As a result, users are strongly encouraged to only use the keychain command with keys.