Demonstrating the Jailhouse Hypervisor Virtualization on the Sitara AM572x Reference Design

TI Design: http://www.ti.com/tool/tidep-0095

Virtualization Jailhouse Hypervisor on AM572x Reference Design

(ACTIVE) TIDEP-0095

Description & Features Technical Documents Support & Training Order Now

View the Important Notice for TI Designs covering authorized use, intellectual property matters and disclaimers.

Key Document

- Virtualization: Jailhouse Hypervisor on AM572x Reference Design (Rev. A) (PDF 540 KB)
  12 Oct 2017 156 views

» View All Technical Documents (9)

Description

Industrial embedded systems are balancing traditional and proven real-time solutions based on bare-metal or real-time operating system (RTOS) with new requirements to add cloud connectivity and advanced graphical interfaces. Linux is often the most efficient way to provide sophisticated and secure cloud connectivity and enable advanced human to machine (HMI) interfaces. Modern embedded processors such as the Sitara AM5728 processor offer the ability to integrate the functionality of a real-time application with Linux applications. This TI design describes using the ARM® Cortex®-A15 cores and an open source static hypervisor called Jailhouse to support the coexistence of real-time and Linux applications.

Features

- Jailhouse Embedded Hypervisor running on Sitara AM572x with Linux on one ARM Cortex-A15 core and bare-metal on the other ARM Cortex-A15 core
- Demonstration of static partitioning of AM572x peripherals between Linux and bare-metal
- Demonstration of support for running a bare-metal binary and an RTOS based binary on the second core
- Performance (interrupt latency) measurements for the virtualized bare-metal system with and without a processing load on Linux
- Tested on the TMDXIDK5728 and TMDSEVM5728 evaluation boards

tideP-0095 Virtualization Jailhouse Hypervisor on AM572x Reference Design Board Image

View available purchase options for designs kits, evaluation modules and/or the bill of materials.

$899.00(USD)

Order Now

View the Important Notice for TI Designs covering authorized use, intellectual property matters and disclaimers.

Schematic/Block Diagram

Quickly understand overall system functionality.

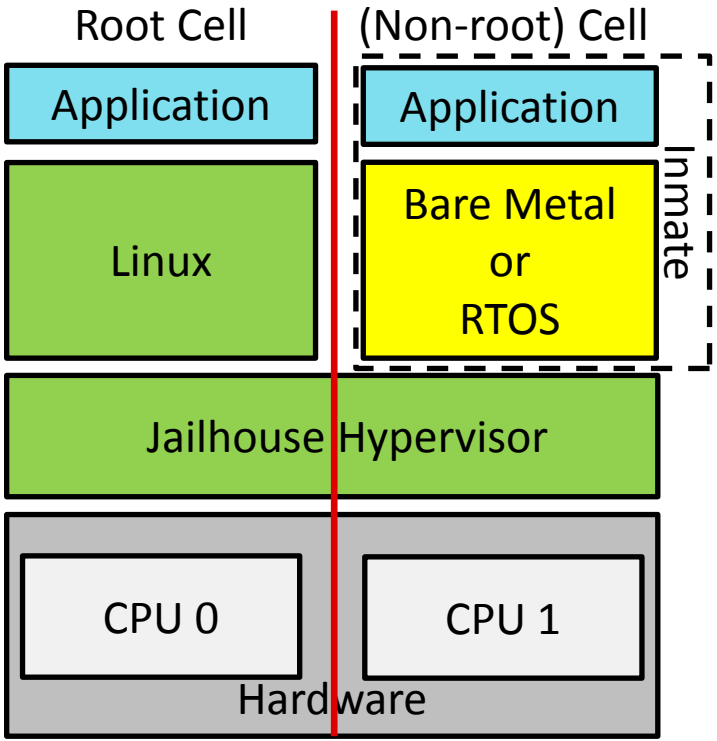
Design Guide

Get results faster with test and simulation data that's been verified.

Design Files

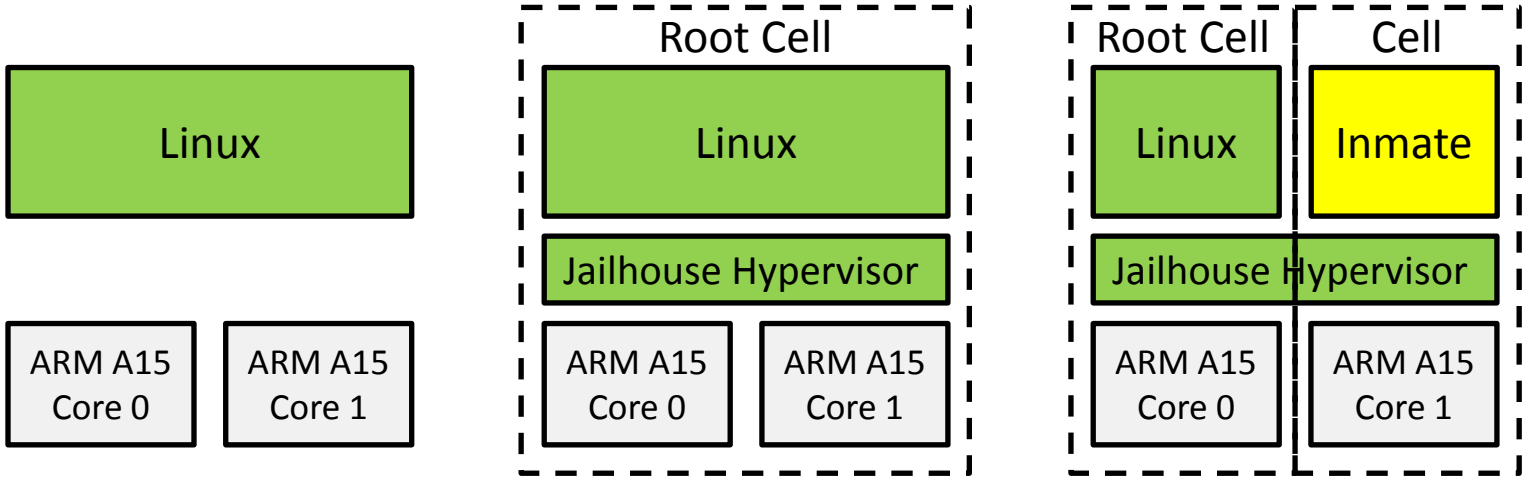
Download ready-to-use system files to speed your design process. Get Viewer.

Jailhouse Overview

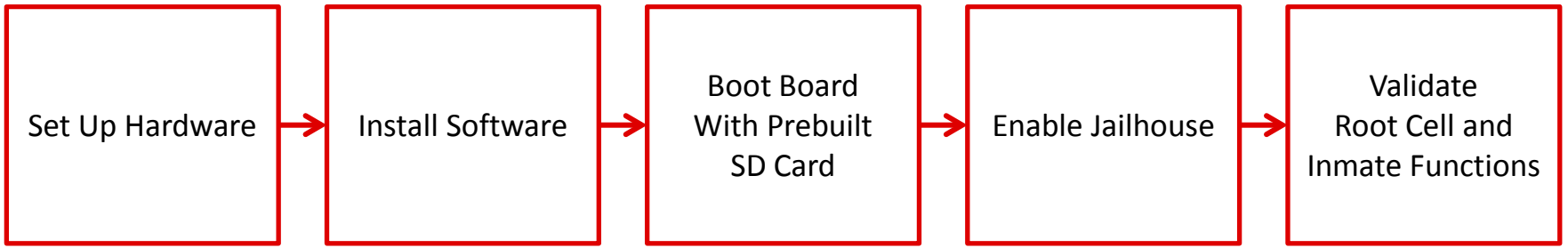


The Jailhouse Linux-based partitioning hypervisor is an open-source project:
https://github.com/siemens/jailhouse

Jailhouse Initialization on AM572x



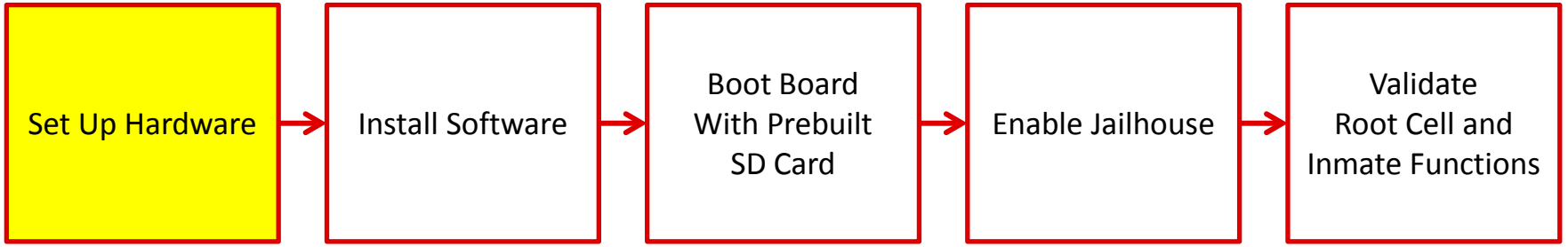
Jailhouse Hypervisor Demonstration Overview



Processor SDK Jailhouse Hypervisor Wiki:

http://processors.wiki.ti.com/index.php/Processor_SDK_Jailhouse_Hypervisor

Set Up Hardware

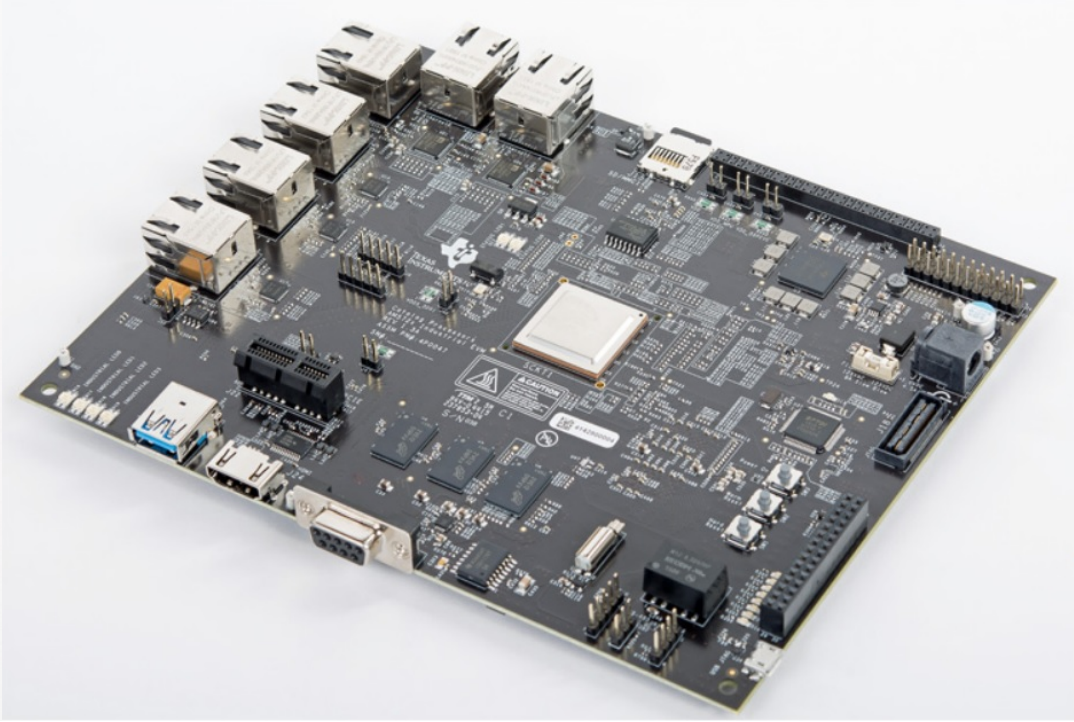


Supported Hardware

AM5728 EVM

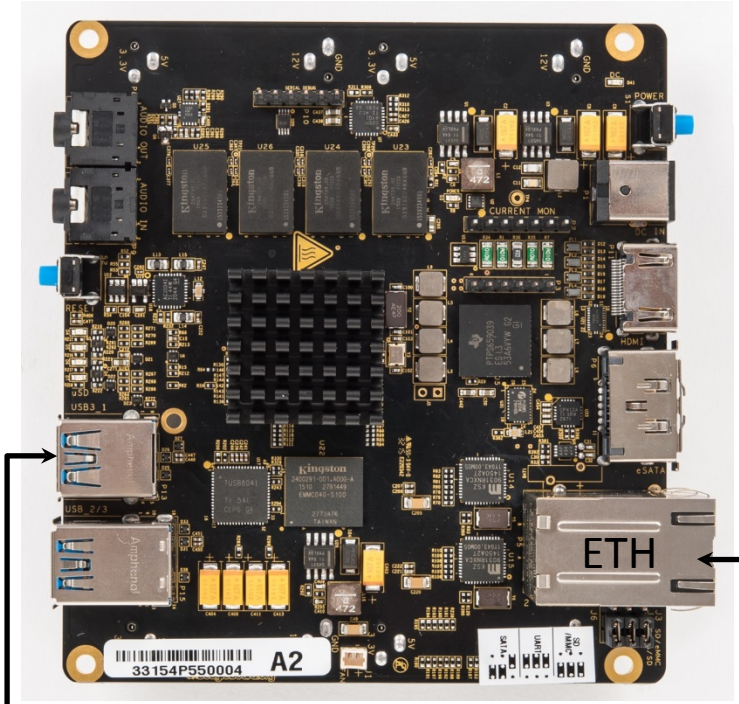
http://www.ti.com/tool/TMDSEVM572X

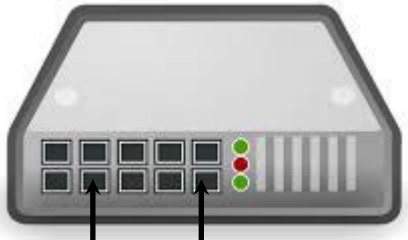
AM5728 IDK



http://www.ti.com/tool/TMDXIDK5728

Set Up Hardware





Host

ETH

USB

Port: COM3

Baud rate: 115200

Data: 8 bit

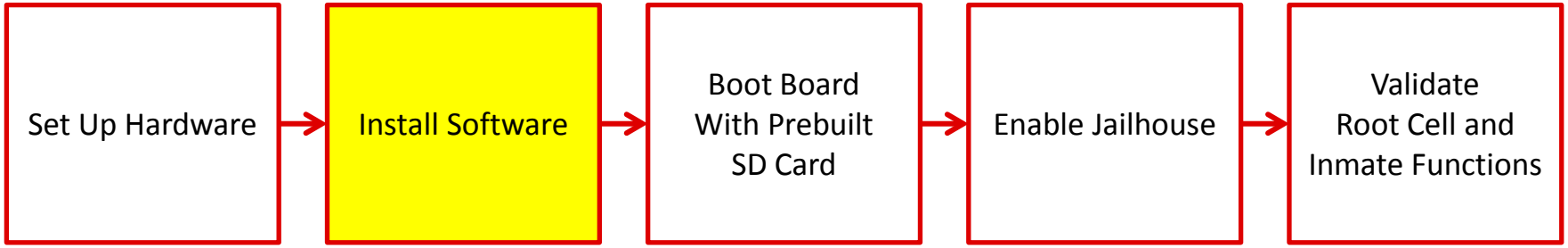
Parity: none

Stop: 1 bit

Flow control: none

Serial

Install Software



Jailhouse Software

- **jailhouse.ko kernel module** is located in the directory:

  /lib/modules/4.9.28-<gitid>/extra/driver

- **jailhouse.bin hypervisor** is located in the directory:

  /lib/firmware

- **Jailhouse management tools** are located in the following directories:

  /usr/local/libexec/jailhouse

  /usr/sbin

Software Setup

The boot arguments need to be modified to use Jailhouse:

- Modify the boot arguments.

  ```
  printenv args_mmc
  ```

- Modify the boot arguments to allocate more virtual memory.

  ```
  setenv args_mmc ${args_mmc} vmalloc=512M
  ```

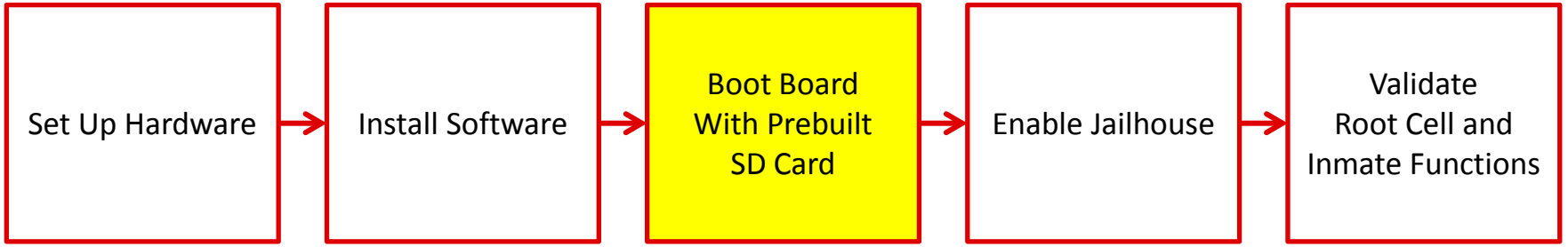
- Modify the boot arguments to identify the correct device tree.

  ```
  setenv findfdt 'setenv fdtfile <device-tree>'
  ```

  where `<device-tree>` is one of the following:

  - `am572x-evm-jailhouse.dtb` for the AM572x EVM
  - `am572x-idk-jailhouse.dtb` for the AM572x IDK

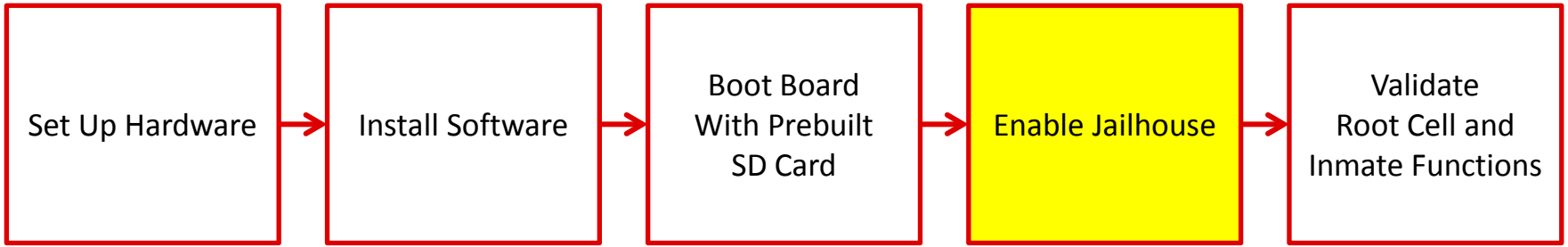
Boot Board With Prebuilt SD Card



- Save the boot arguments and boot the board.

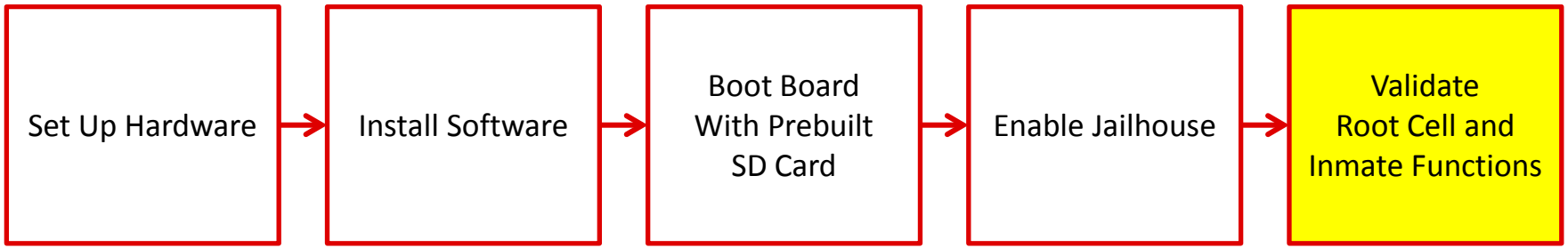
```
saveenv args_mmc
boot
```

Enable Jailhouse

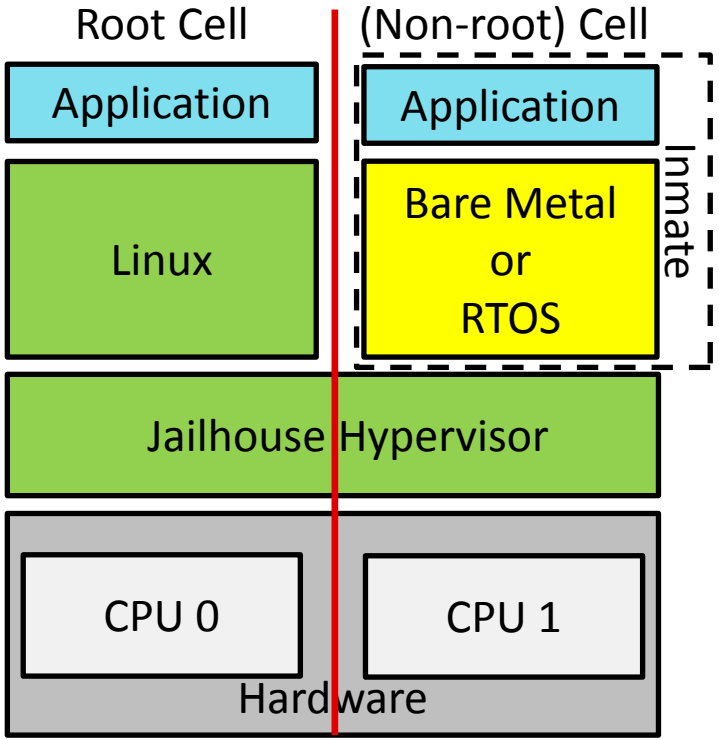


- Insert the kernel module:
  `modprobe jailhouse`
- Enable the hypervisor:
  `jailhouse enable /usr/share/jailhouse/examples/am57xx-evm.cell`
- Create a cell for the inmate:
  `jailhouse cell create /usr/share/jailhouse/examples/am57xx-evm-ti-app.cell`
- Load the bare metal binary:
  `jailhouse cell load 1 /usr/share/jailhouse/examples/ti-app.bin`
- Start the binary:
  `jailhouse cell start 1`

Validate Root Cell and Inmate Functions



Validate Root Cell Function



For More Information

- Virtualization: Jailhouse Hypervisor on AM572x Reference Design: http://www.ti.com/tool/tidep-0095
- Sitara Processors Product Overview: http://www.ti.com/sitara
- AM572x Evaluation Module: http://www.ti.com/tool/tmdsevm572x
- AM572x Industrial Development Kit IDK): http://www.ti.com/tool/TMDXIDK5728
- Processor SDK for AM57x Sitara Processors: http://www.ti.com/tool/processor-sdk-am57x
- Processor SDK Jailhouse Hypervisor Wiki: http://processors.wiki.ti.com/index.php/Processor_SDK_Jailhouse_Hypervisor
- For questions about this training, refer to the E2E Community Forums for Sitara Processors at http://e2e.ti.com/support/arm/sitara_arm/f/791/t/277411