LDD EXAM

EnsiCaen – SATE 3A

2019/2020

Read the attached Linux device driver.

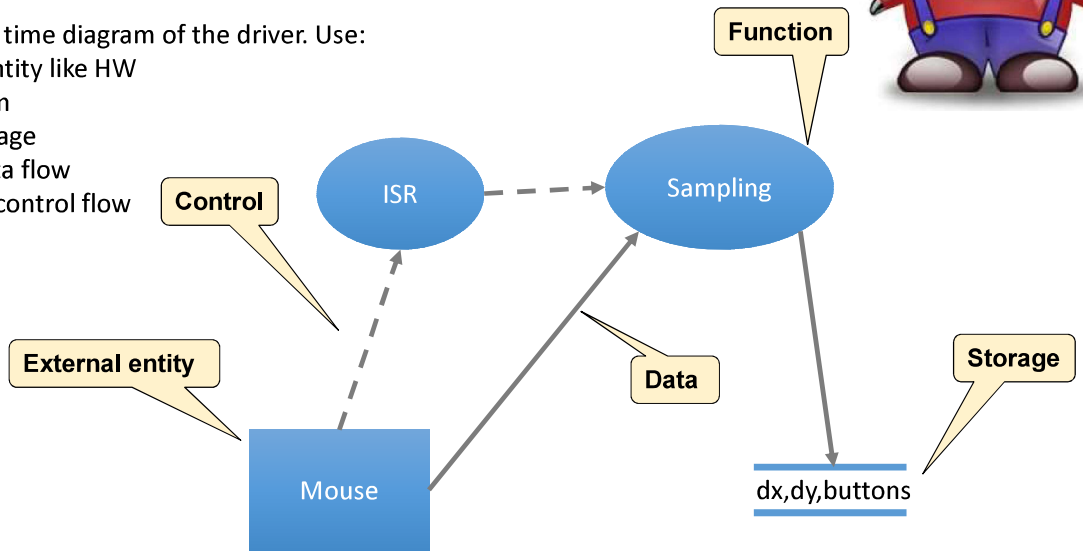
The source has been simplified to reduce printing for this exam.

Write and draw the SART view of this driver as explained next slide.

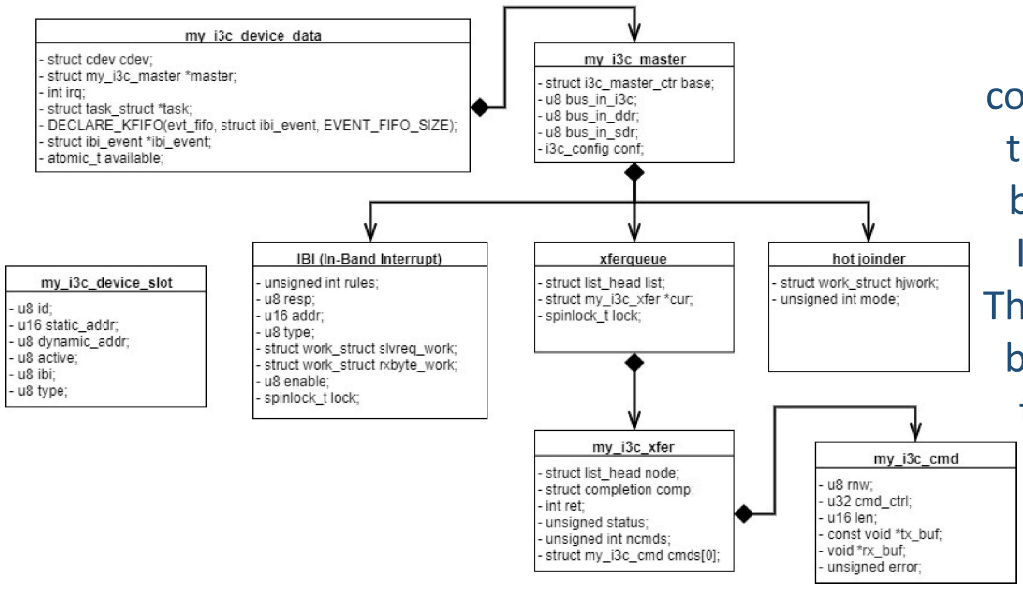
SART - Data flow and real time example

Provide data flow and real time diagram of the driver. Use:

- Box for external entity like HW
- Bubble for function
- Stack for data storage
- Plain arrow for data flow
- Dashed arrow for control flow



I3C Device Driver Data Structure Breakdown



For your convenience, here is the data structure breakdown of the I3C device driver. This may help you to better understand the source code.

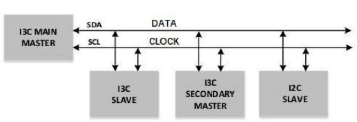
I3C Protocol

Extra Documentation

- MIPI standardized protocol designed to overcome I2C limitations (limited speed, external signals needed for interrupts, no automatic detection of the devices connected to the bus, …) while remaining power-efficient

MIPI I3C Fact Sheet

MIPI I3C Standardized Sensor Interface



Two-wire communication interface, clock (SCL) and data (SDA)

Number of gates	< 2,000
Bandwidth	> 33 Mbps
Features	In-band interrupts In-band command codes Dynamic addressing Multi-master / multi-drop Hot-join support Backward compatible with I²C

- Low-power and space efficient design intended for mobile devices (smartphones and IoT devices.)
- Two-pin interface that is a superset of the I2C standard. Legacy I2C slave devices can be connected to the newer bus.
- In-band interrupts over the serial bus rather than requiring separate pins
- Standard Data Rate (SDR) throughput up to 12.5 Mbit/s using CMOS I/O levels,
- High Data Rate (HDR) modes permitting throughput comparable to SPI while requiring only a fraction of I2C Fast Mode power.[16]
- A standardized set of common command codes
- Command queue support
- Error Detection and Recovery (parity check in SDR mode and 5bit CRC for HDR modes)
- Dynamic address assignment (DAA) for I3C slaves, while still supporting static addresses for I2C legacy devices
- I3C traffic is invisible for legacy I2C devices when equipped with I2C spike filters, achieved by SCl HIGH times of less than 50ns
- Hot-join (some devices on the bus may be powered on/off during operation)
- Multi-master operation with well-defined hand-off

ENSI CAEN

I3C Transfer Type

Extra Documentation

- I3C defines 3 different classes of transfer in addition to I2C transfers which are here for backward compatibility with I2C devices.
- **I3C CCC commands**
  - CCC (Common Command Code) commands are meant to be used for anything that is related to bus management and all features that are common to a set of devices.
- **I3C Private SDR transfers**
  - Private SDR (Single Data Rate) transfers should be used for anything that is device specific and does not require high transfer speed.
- **I3C HDR commands**
  - HDR commands should be used for anything that is device specific and requires high transfer speed.