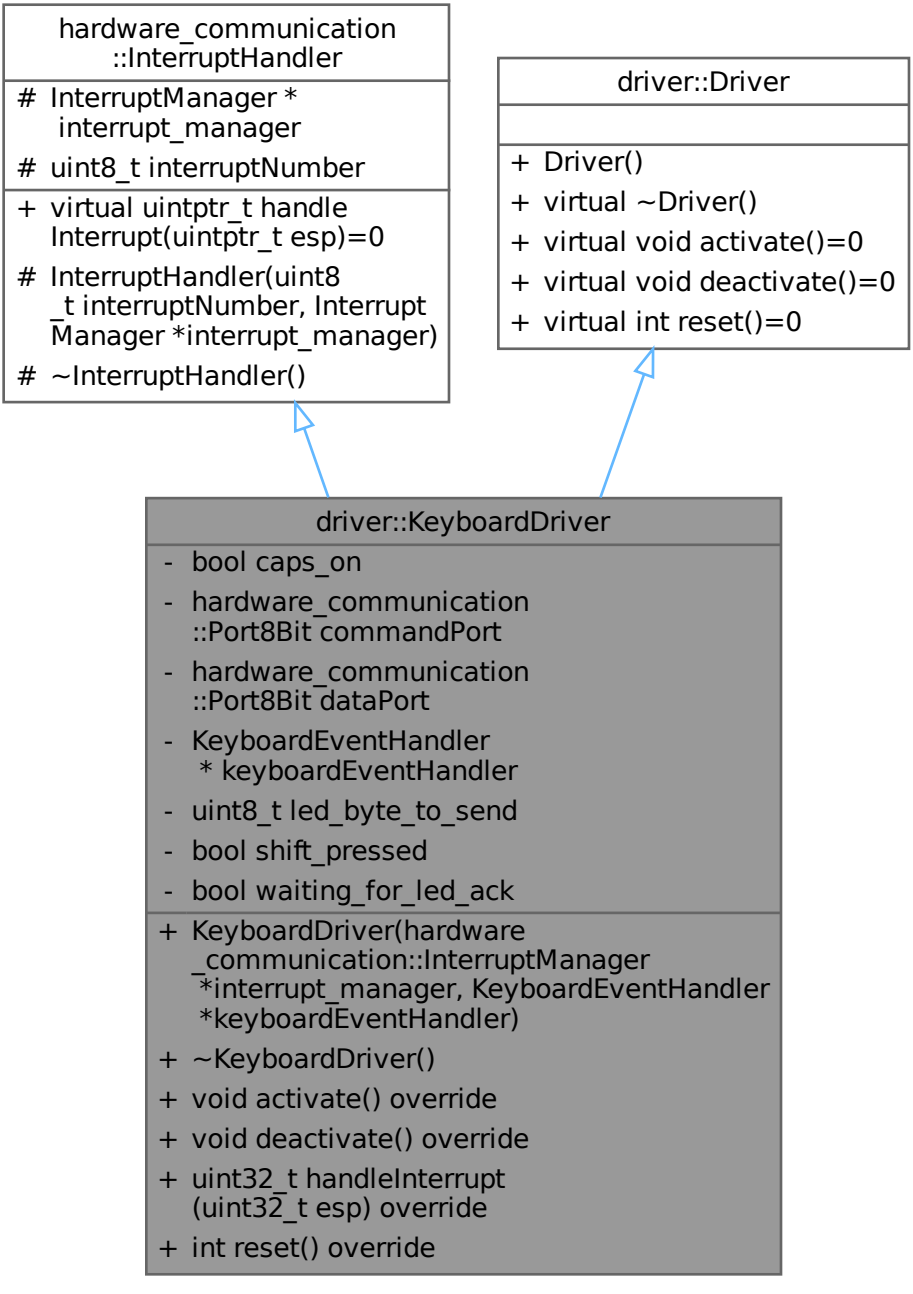
hardware_communication
::InterruptHandler
InterruptManager *
interrupt_manager
uint8_t interruptNumber
+ virtual uintptr_t handle
Interrupt(uintptr_t esp)=0
InterruptHandler(uint8
_t interruptNumber, Interrupt
Manager *interrupt_manager)
~InterruptHandler()
driver::Driver
+ Driver()
+ virtual ~Driver()
+ virtual void activate()=0
+ virtual void deactivate()=0
+ virtual int reset()=0
driver::KeyboardDriver
- bool caps_on
- hardware_communication
::Port8Bit commandPort
- hardware_communication
::Port8Bit dataPort
- KeyboardEventHandler
* keyboardEventHandler
- uint8_t led_byte_to_send
- bool shift_pressed
- bool waiting_for_led_ack
+ KeyboardDriver(hardware
_communication::InterruptManager
*interrupt_manager, KeyboardEventHandler
*keyboardEventHandler)
+ ~KeyboardDriver()
+ void activate() override
+ void deactivate() override
+ uint32_t handleInterrupt
(uint32_t esp) override
+ int reset() override