

# Mesih Veysi Kılınç

<https://github.com/MesihK>

mesihkilinc@gmail.com

+90-543-864-2148

I am an Electronic Engineer doing Master in Computer Science. Interested in Linux Kernel and love doing embedded related works. Fascinated by programming and started to code during the first year of high school, doing it since then.

## OPEN SOURCE CONTRIBUTIONS - LINUX KERNEL

---

- **Applied:** initial support for "suniv" Allwinner new ARM9 SoC
- Add support for DMA and audio codec of F1C100s
- Timer & SPI support for Allwinner suniv F1C100s

## EDUCATION

---

- **Gebze Technical University** Kocaeli, Turkey  
*Master of Science in Computer Science; GPA: 4.00/4.00* Feb. 2018 – Present
  - **Scientific preparation:** Successfully finished 5 courses under scientific preparation to Computer Science which are: *Object Oriented Programming, Data Structures, Operating Systems, Computer Architecture, Discrete Mathematics*
- **Istanbul Technical University** Istanbul, Turkey  
*Bachelor of Engineering in Electronics and Communication; GPA: 2.71/4.00* Sep. 2011 – Feb. 2016

## EXPERIENCE

---

- **Gebze Technical University** Kocaeli, Turkey  
*Research Assistant* Feb 2018 - Present
  - **Haptic Delta Robot:** Designed a hardware to control delta robot. Developed software to try control techniques on device.
  - **Swarm Unmanned Aerial Vehicle:** Designed hardware and developed software for a UAV telemetry modem and a RSSI Sensor which can measure the strength of RF signals in specified band.
- **Otokar** Sakarya, Turkey  
*Software Engineer* August 2016 - Feb 2018
  - **Border Surveillance and Reconnaissance Vehicle:** We were developing management software for the radar of vehicle. At some point no matter how hard we tried we couldn't solve problems. When we try to implement new feature, old ones started to broke. We had tremendous time pressure. I stepped back, and decided to change whole core code. Introduced a hierarchical state machine that manages the radar. Since states managed correctly, code size shrunk and started to behave correctly. Later on all other devices that used in this vehicle platform revised by me to use state machines. (**Embedded, C++, Qt, Yocto**)
  - **CLI Simulator:** It was very hard to try all changes, to make sure we didn't broke anything, because we used actual hardware. I decided to write a simulator for radar to understand radar correctly and fix bugs. This made development fast and allowed to find long lasting bugs. (**Python**)
- **CTech** Istanbul, Turkey  
*Hardware Engineer* July 2015 - August 2016
  - **UAV Modem:** I met Linux with this project. A hardware was given to me to write device driver. Suprised to find that the driver already existed, and found that with little tweaking it just started to work. More suprising thing was how powerful the shell was. It was so easy to connect little things to create a powerful beast that uses my hardware without even writing a line of code. (**Linux Kernel, Device Drivers**)
  - **Linux Training:** Took Linux training from Nazim Koc (uCanLinux.com). Gained knowledge on **U-Boot, Linux Kernel, Busybox compilation and building RootFS.**

## MAJOR PROJECTS

---

- **Real-time Video Compression Device:** Coded software for an H264 video compression device. It is based on IMX6 SoC and also uses real-time patched Linux and GStreamer. Build the required device tree and userspace software that is based on GStreamer (**Gstreamer, Yocto**)
- **Voice Transceiver Device:** A device capable of recording and sending voice at ISM band. It is an embedded Linux system that utilizes Opus codec and Alsa. Designed and produced the hardware and coded necessary **ALSA SoC driver, SoC's DMA Driver, RFIC device driver** and userspace software. Build the device tree and RootFS.
- **Cansat Competition, Abilene TX:** Designed hardware and coded software according to requirements of the competition. It uses Cortex-M based MCU.
- **Turksat 3USAT Cube Satellite, ITU:** Coded software for onboard computer of the satellite.