

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería Electrónica



---

# Bitácora Proyecto 2 Taller de Sistemas Embebidos

---

Taller de Sistemas Embebidos

Integrantes:

David Monge Naranjo

Profesor:

Dr. Ing. Johan Carvajal Godínez

10 de noviembre de 2023

11 DE OCTUBRE DEL 2023

**AVANCES:**

- Me reuní con Jordani para definir cómo nos vamos a organizar para trabajar para el proyecto.
- Decidimos trabajar los 2 juntos y no repartir el trabajo, entonces empezamos directamente hoy con el trabajo.
- Adelantamos bastantes diagramas y el documento con la propuesta de diseño.

11 DE OCTUBRE DEL 2023

**AVANCES:**

- Nos reunimos Jordani y yo para continuar con la propuesta de diseño, avanzamos bastante, queda revisar y pulir el documento pero en términos generales ya quedó listo.

22 DE OCTUBRE DEL 2023

**AVANCES:**

- Me reuní con Jordani para empezar a hacer con la parte práctica, creamos un nuevo proyecto de yocto y agregamos los layer de meta-raspberrypi, meta-tensorflow y meta-java que es una dependencia de tensorflow, también de openembedded meta-oe, meta-networking, meta-multimedia y meta-python. Los comandos son estos, teniendo ya los repositorios clonados y el proyecto creado:

```
1 bitbake-layers add-layer meta-raspberrypi
2 bitbake-layers add-layer meta-openembedded/meta-oe
3 bitbake-layers add-layer meta-openembedded/meta-python
4 bitbake-layers add-layer meta-openembedded/meta-networking
5 bitbake-layers add-layer meta-openembedded/meta-multimedia
6 bitbake-layers add-layer meta-tensorflow
```

- Tuvimos un error con tensorflow al crear la imagen *base*, pero quitando esta dependencia, el proceso del bitbake funcionó y se generó la imagen.

31 DE OCTUBRE DEL 2023

**AVANCES:**

- Me reuní con Jordani para intentar flashear la raspberri, usamos de base este video Building embedded GNU/Linux distribution for Raspberry Pi using the Yocto Project. Los comandos fueron estos para cargar en la SD:

```
1 sudo umount /dev/sda*
2 sudo dd if=../cursos/emababidos/proyecto2/core-image-base-
raspberrypi2.rpi-sdimg of=/dev/sda bs=4M
3 sync
```

- Con esto, colocando la SD en la raspberry y conectando teclado y un monitor pudimos correr la imagen, pero tuvimos errores al usar opencv, da este error:

```
1 Required baseline features:
2 ID=100 (NEON) - NOT AVAILABLE
3 terminate called after throwing an instance of 'cv::Exception'
4 what(): OpenCV(4.5.5) /usr/src/debug/opencv/4.5.5-r0/git/modules/
core/src/system.cpp:677: error: (-215:Assertion failed) Missing
support for required CPU baseline features. Check OpenCV build
configuration and required CPU/HW setup. in function 'initialize'
```

- También intentamos usar emacs pero da este error:

```
1 emacs: Could not open file: /dev/tty
```

02 DE NOVIEMBRE DEL 2023

#### AVANCES:

- Jordani y yo le pedimos ayuda a David Duarte para ver los errores de tensorflow y opencv, fue una reunión corta pero nos ayudó y guió bastante, nos dio algunas guías para flashear la raspberry.
- También luego, en la máquina virtual de Jordani estuvimos intentando montar la imagen de yocto pero estuvieron dando algunos problemas el bitbake, así que se decidió reiniciar el proyecto.
- El error que estaba dando es este:

```
1 ERROR: Unable to start bitbake server (None)
2 ERROR: Server log for this session (/home/jordanimejia/yocto/poky/
build/bitbake-cookerdaemon.log):
3 --- Starting bitbake server pid 28603 at 2023-11-02
05:26:21.534726 ---
4 ERROR: Layer gstreamer1.0 is not comparable with the core layer
which only supports these series: dunfell (layer is compatible
with thud warrior zeus)
```

03 DE NOVIEMBRE DEL 2023

#### AVANCES:

- Usé este modelo de base para reconocer las expresiones faciales Emotion-detection.
- Adapté el código para la aplicación propia, donde no se muestra una ventana con el video en tiempo real, sino más bien cada cierto intervalo se guarda una imagen con el recuadro y la emoción detectada, además de un archivo de texto plano con el nombre de la imagen y la emoción. Por otro lado, también agregué la funcionalidad de usar un archivo .json para leer una pequeña configuración, el intervalo de tiempo de cada toma de dato y un booleano que indica si cerrar o no la aplicación.
- Usé tensorflow lite para crear el modelo en este formato y en el código cambié todo

para usar este formato.

04 DE NOVIEMBRE DEL 2023

**AVANCES:**

- Me reuní con Jordani para discutir el progreso hasta el momento, ya él tiene más o menos avanzada la construcción de la imagen y el uso de las librerías de python en la imagen de yocto.
- También empezamos a crear la interfaz gráfica para la aplicación de la computadora, vamos a usar la librería PySimpleGUI.

05 DE NOVIEMBRE DEL 2023

**AVANCES:**

- Seguí trabajando un poco en la interfaz gráfica, no hice mucho pero probé como mostrar una imagen.

06 DE NOVIEMBRE DEL 2023

**AVANCES:**

- Pude mostrar correctamente las imágenes e ir las pasando una por una, también puse un botón para cerrarla.
- Investigué un poco sobre la librería de paramiko para conectar por ssh.

07 DE NOVIEMBRE DEL 2023

**AVANCES:**

- Integré en la aplicación la funcionalidad para conectar y desconectar por ssh.
- Pude también conectando con el computador local ejecutar la aplicación que recolecta emociones, el comando que usé fue este:

```
1 a, b, c = client.exec_command('/home/david/anaconda3/condabin/  
conda run -n emotion_env bash -c "cd /home/david/cursos/embabidos/  
proyecto2/proyecto2_TSE/emociones/ && python3 /home/david/cursos/  
embabidos/proyecto2/proyecto2_TSE/emociones/emotions.py"')
```

- Tengo un ambiente de conda dedicado para el proyecto, en la raspberry no se necesita más que ejecutar el comando de python3 directamente.
- Avancé con las funciones para usar la rasp por ssh, aún quedan pendientes algunas, pero estoy teniendo problemas con las expresiones regulares.

08 DE NOVIEMBRE DEL 2023

**AVANCES:**

- Terminé la funcionalidad de conexión ssh en la aplicación local, esto lo depuré usando la conexión con la dirección local de mi computador.
- También hice algunos cambios en la interfaz para que al presionar el botón de encender aplicación, este cambie su funcionalidad y su mensaje para poder apagar la aplicación más bien.
- Realizando esto no tuve ningún problema en especial, revisando documentación de las librerías de Paramiko y de PySimpleGUI fue bastante fluído el trabajo.

09 DE NOVIEMBRE DEL 2023

**AVANCES:**

- Este día me reuní con Jordani para verificar el funcionamiento de la conexión ssh de la raspberry.
- Investigamos un poco rápidamente el procedimiento para hacer esto.
- Dentro de lo que encontramos es que para obtener la IP del dispositivo se debe ejecutar este comando en la consola, con permisos de root:

```
1 ifconfig
```

- Para realizar la conexión realizamos la creación de un nuevo usuario en la raspberry, usamos este comando:

```
1 adduser pi
```

- Esto crea un usuario llamado *pi*, va a solicitar una contraseña, colocamos *pi* igualmente.
- Luego de esto tuvimos un problema, ya Jordani había verificado que la aplicación funcionaba correctamente en la raspberry, pero no funcionaba en el nuevo usuario. Esto se solucionó dando acceso de lectura, escritura y ejecución al usuario para la carpeta `/usr/bin` y para el dispositivo `/dev/video0`.
- Con esto el proyecto queda listo, todo funciona correctamente. :)

## Referencias