



**FACULTAD
DE INGENIERIA**

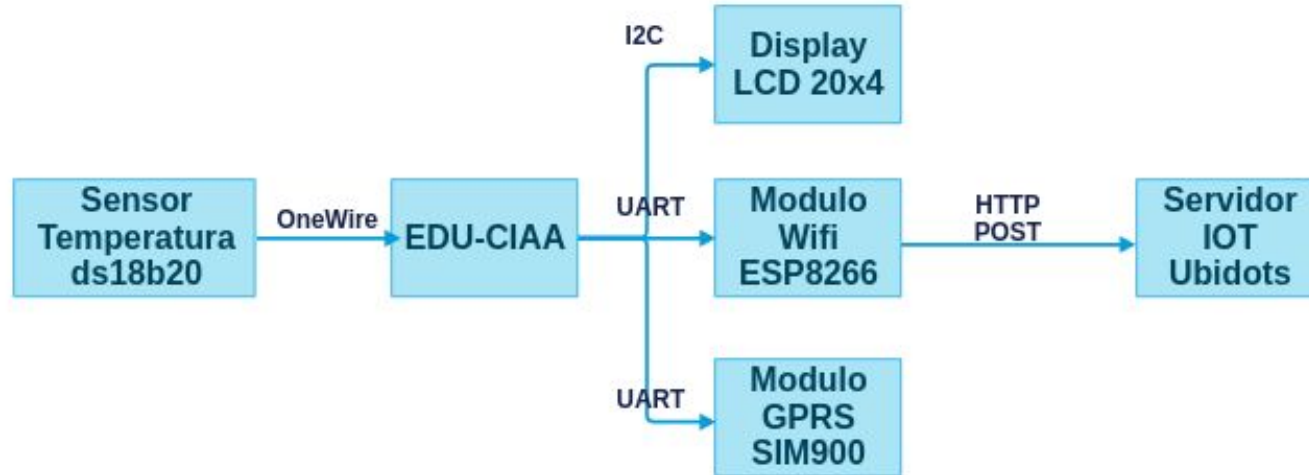
Universidad de Buenos Aires

PROTOSCOLOS DE COMUNICACIÓN EN
SISTEMAS EMBEBIDOS

Presentación Trabajo Final

Matias Alvarez

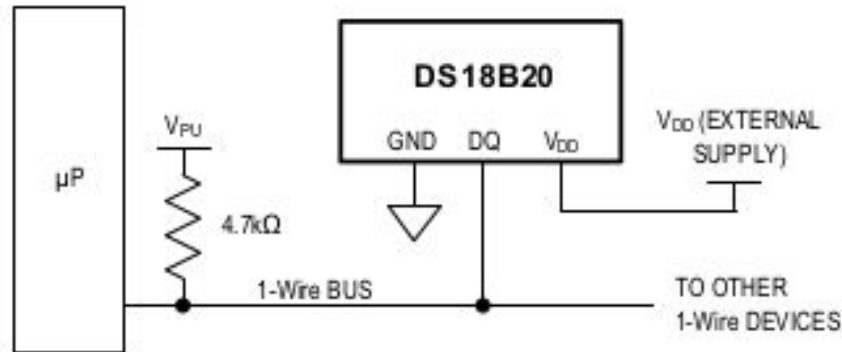
DESCRIPCIÓN TÉCNICO - CONCEPTUAL



- **Protocolo One Wire**
- **Protocolo I2C**
- **Protocolo Wifi - TCP - HTTP(POST)**

SENSOR DE TEMPERATURA DS18B20

- Protocolo One Wire - Un único cable de datos.
- Rango -55°C a 125°C , con error $\pm 0,5^{\circ}\text{C}$.
- Resolución programable de 9 a 12 bits.
- 64 bit ROM.
- EEPROM con 3 registros de configuración.
- SRAM Scratchpad de 9 bytes.

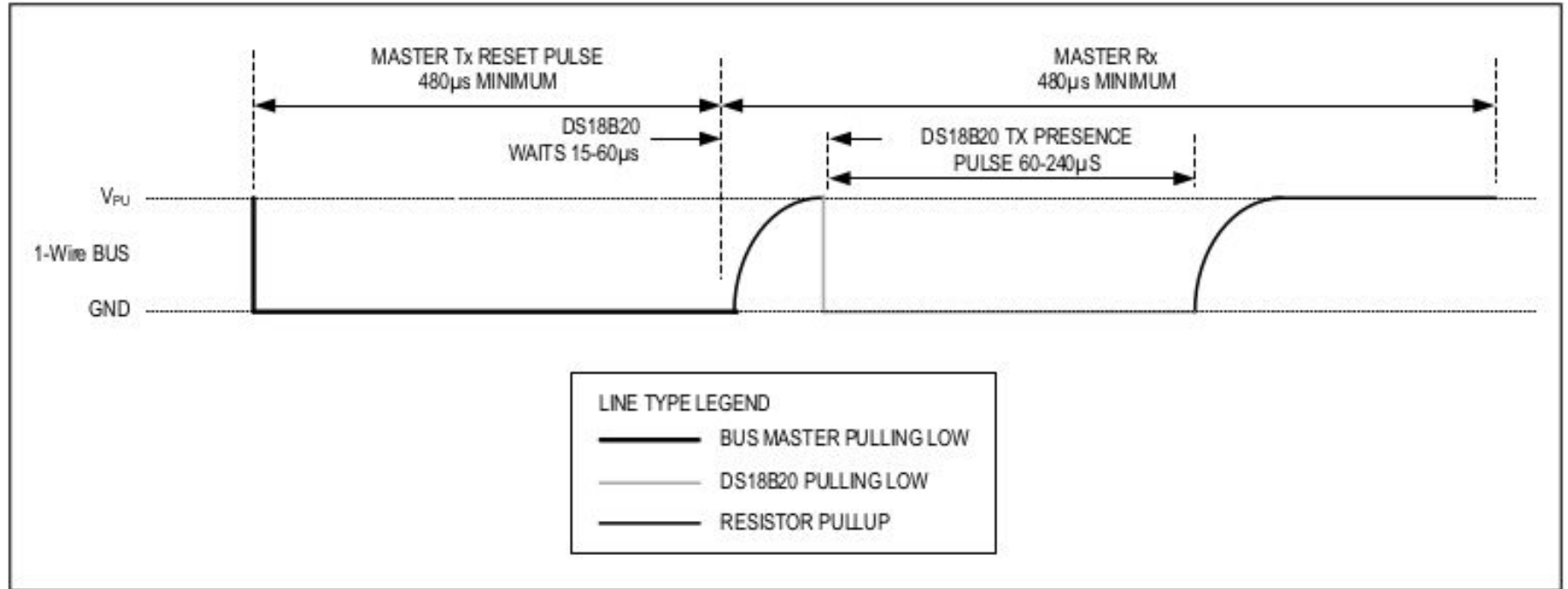


SECUENCIA DE TRANSACCIÓN

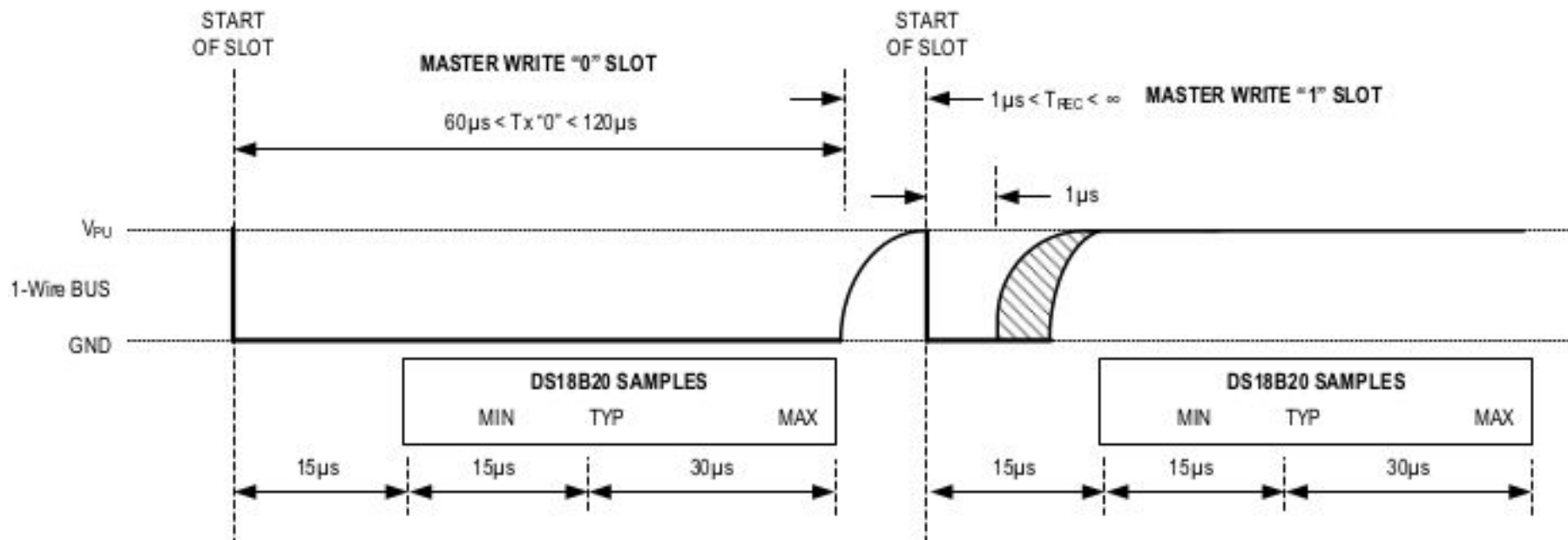
Teniendo en cuenta que el sensor es siempre esclavo, los pasos para una transacción de datos son los siguientes:

- 1. Inicialización**
- 2. Comando ROM**
- 3. Comando Función ds18b20**

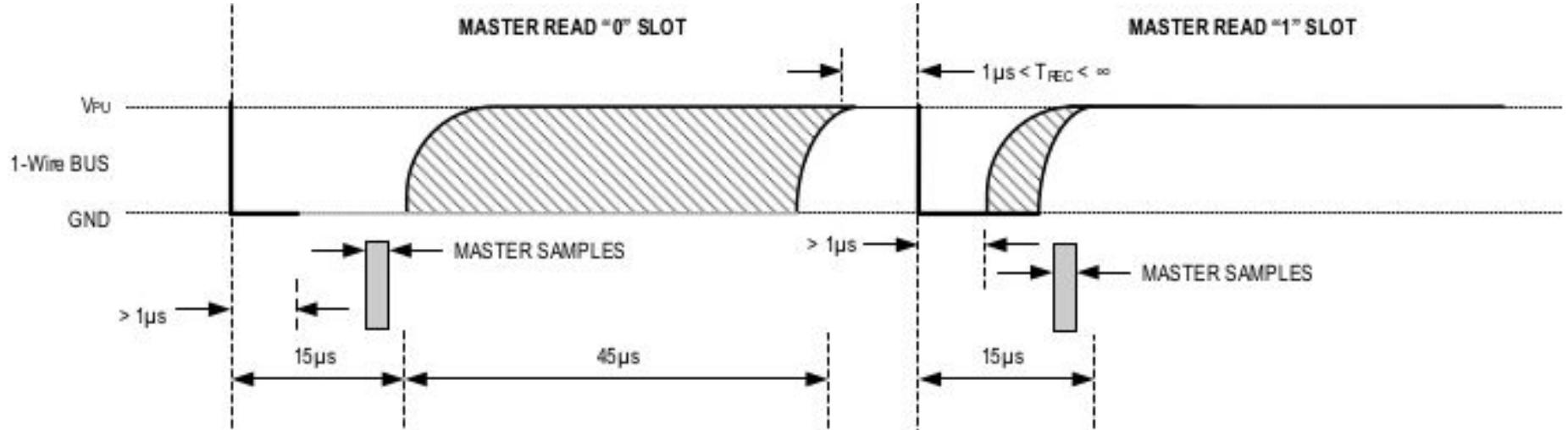
PROTOCOLO ONE WIRE - INICIALIZACIÓN



PROTOCOLO ONE WIRE - ESCRITURA



PROTOCOL ONE WIRE - LECTURA



PROTOCOLO ONE WIRE - MIDIENDO LA TEMPERATURA

**SCRATCHPAD
(POWER-UP STATE)**

BYTE 0	TEMPERATURE LSB (50h)	} (85°C)
BYTE 1	TEMPERATURE MSB (05h)	
BYTE 2	T _H REGISTER OR USER BYTE 1*	
BYTE 3	T _L REGISTER OR USER BYTE 2*	
BYTE 4	CONFIGURATION REGISTER*	
BYTE 5	RESERVED (FFh)	
BYTE 6	RESERVED	
BYTE 7	RESERVED (10h)	
BYTE 8	CRC*	

EEPROM

T _H REGISTER OR USER BYTE 1*
T _L REGISTER OR USER BYTE 2*
CONFIGURATION REGISTER*

Secuencia:

- 1. SKIP ROM**
- 2. CONVERT T**
- 3. SKIP ROM**
- 4. READ
SCRATCHPAD**

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2 ³	2 ²	2 ¹	2 ⁰	2 ⁻¹	2 ⁻²	2 ⁻³	2 ⁻⁴
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2 ⁶	2 ⁵	2 ⁴

S = SIGN

UBIDOTS



UBIDOTS

1. **Educational Users vs Business Users**
2. **Token: Clave temporal para ser utilizada en los requerimientos al servidor.**
3. **API Key: Clave única utilizada para generar los tokens.**
4. **Device Label : Identificador de dispositivo.**
5. **Variable Label : Identificador de variable dentro de un dispositivo.**

UBIDOTS - ENVIANDO VALORES

La forma más sencilla es especificar el Device Label en la URL y realizar un POST.

El cuerpo de HTTP debe ser un diccionario de JSON donde cada clave corresponde a un Variable Label.

```
POST /api/v1.6/devices/{LABEL_DEVICE}/?token={TOKEN} HTTP/1.1
```

```
Host: things.ubidots.com
```

```
Content-Type: application/json
```

```
Content-Length: 76
```

```
{"temperature": 10, "luminosity": {"value":10}, "wind_speed": [{"value": 11,  
"timestamp":10000}, {"value": 12, "timestamp":13000}]}
```

¿Preguntas?

Muchas Gracias