EVSE CHARGING BOARD PROJECT REQUIREMENTS AND MCU CHOICE

Compiled By : Madhav Maheshwari
Mentor : Pushkar Suthar

Table Of Contents :

Project Aims and Requirements 3
Choosing The Microcontroller 3

Project Aims and Requirements

The project aims to develop a simplified version of the EVSE charging board, containing all the essential functionality of such a circuit.The board is supposed to have the following functionality.

1) Control Pilot Signalling circuits and proper charging state interpretation,the ability to generate the 12V control pilot signal while also implementing detection circuits to allow for identifying state using an MCU.
2) Current and Voltage metering, measuring the amount of power being consumed.
3) Protection circuitry such as GFCI and Leakage Current Protection
4) A temperature detection circuit to handle overheating situations.
5) Power Supply with flyback topology for efficient conversion of Mains power line voltage to suitable voltage levels for powering control electronics.
6) A main relay to regulate supply of power to the battery.
7) A display to show charging parameters, such as power being consumed,charging status

All of these functionalities are to be implemented in adherence to the standards.

Choosing The Microcontroller

The STM32 microcontroller by STMicroelectronics has been chosen as the heart of the project.Its comprehensive documentation, online tutorials and Coding IDE will make the development simpler.Although the choice of a particular variant of this MCU, has to be made in accordance to the project requirements.

1) ADC: For current and voltage metering, we will likely require a MCU with a high ADC resolution and sampling rate so as to maintain accuracy in our measurements.Since we will be working with only single phase supplies, we only require a single analog pin for this purpose.

   In addition to the current and voltage metering the ADC will also be required for control pilot voltage detection and temperature measurement.
   So, as a total 3 ADC channels will be required for the project with high resolution and sampling rate.

2) Timer : A single timer capable of generating a stable 1 Khz pulse with a variable duty cycle will be required for the Control Pilot PWM signal generation.

3) I2C : I plan to use a 16x2 LCD display for the purpose of showing the charging state and power being consumed.For that the MCU will have to possess I2C capability.

4) Digital IO : Digital IO pins will be required for controlling the main relay and for the GFCI or leakage protection. Thus it is preferred for the MCU to have around 5-6 digital IO pins.

5) Interrupt Capability will be needed to respond to charging fault conditions such as overvoltage,undervoltage and leakage current.

6) A watchdog timer to improve reliability in case of firmware failures.

7) UART/USART for programming and debugging.

Based on these requirements a comparison of the 3 most fitting MCUs has been given below:

Feature	STM32F103C8T6 (Blue Pill)	STM32L432KC (L4 series)	STM32H743ZI (H7 series)
Core	Cortex-M3 @ 72 MHz	Cortex-M4 @ 80 MHz (FPU)	Cortex-M7 @ 400 MHz (FPU + DSP)
ADC Resolution	12-bit	Up to 16-bit (oversampling)	Up to 16-bit (oversampling)
ADC Speed	~1 MSPS	Up to 5 MSPS (12-bit)	Up to 3.6 MSPS (12-bit)
Timers (PWM)	Yes (TIM2–4)	Yes (advanced timers)	Yes (high-end timers)
I2C / UART	Yes	Yes	Yes
Watchdog	Yes	Yes	Yes
Low Power	❌	✅ Ultra-low power	❌ High power
Cost	✅ Very low (~₹100–150)	~₹350–450	❌ High (~₹800–1000)
Dev Board	Blue Pill	Nucleo-L432KC	Nucleo-H743ZI
EMI / Noise Immunity	Moderate	✅ Better analog & digital separation	✅ Excellent
Future Expandability	Limited	Moderate	✅ High (audio, DSP, ML, etc.)

We can also compare these on the basis of RAM and ROM that will be required for our project :

We will require ROM for :
1) Firmware code (PWM control, ADC reading logic, GFCI, protection logic)
2) Libraries for LCD (I2C), ADC calibration, math (e.g., averaging, RMS)
3) Startup code, interrupt vector table, etc.

This leads to an estimated ROM requirement of **32–64 KB minimum**, more is better if you use HAL drivers, printf/debugging, and LCD code.

We will require RAM for :
1) ADC data buffers (for voltage, current, temp sampling)
2) Display buffer (for I2C LCD)
3) State machines and control logic
4) Communication stacks (UART, possibly I2C interrupt-based)
5) Fault-handling and debouncing (relay, GFCI input)

The estimated RAM requirement is **8–16 KB minimum**, more is better for flexibility/debugging.

Now comparing the 3 MCU on the basis of RAM available we get :

MCU	Flash (ROM)	RAM	Notes
STM32F103C8T6	64 KB	20 KB	Bare minimum for your project. Might run out of RAM if using HAL + LCD
STM32L432KC	256 KB	64 KB	Ideal. Plenty of room for code, buffers, HAL, and expansion
STM32H743ZI	2 MB Flash	1 MB	Overkill for now, great if doing advanced UI, networking, ML later

Due to its higher ADC resolution and sufficient RAM and ROM, the STM32L432KC comes out as an optimal choice for such an application. But the cost of this is on the higher side.It will be better if we can find a cheaper alternative.

.

We will be looking at the STM32F401 microcontroller. Given below is a comparison of STM32L432KC and STM32F401 across parameters relevant to this project.

Feature	STM32L432KC	STM32F401CCU6	Remarks
Core	Cortex-M4 @ 80 MHz with FPU	Cortex-M4 @ 84 MHz with FPU	Similar performance, F401 is slightly faster
Flash (ROM)	256 KB	256 KB	✅ Both are equally sufficient for full HAL + LCD usage
RAM	64 KB	64 KB	✅ Both provide ample RAM for buffers and HAL
ADC Resolution	12-bit native, up to **16-bit via oversampling**	12-bit native, no oversampling to 16-bit	🔸 L432 has better ADC resolution flexibility
ADC Speed	Up to 5 MSPS (12-bit)	Up to 2.4 MSPS (12-bit)	🔸 L432 has higher ADC sampling speed
No. of ADC Channels	10	16	✅ Both are more than enough for 3-channel use
PWM (Timers)	TIM1 (Advanced), TIM2, TIM6, TIM21, etc.	TIM1 (Advanced), TIM2, TIM3, TIM4, TIM5, etc.	✅ Both support 1 kHz PWM easily
I²C	I²C1, I²C3	I²C1, I²C2	✅ Both sufficient for I²C LCD interface
UART / USART	USART2, LPUART1	USART1, USART2	✅ Both offer debugging/programming options
Digital IOs	~21 GPIOs	~37 GPIOs	✅ F401 offers more expansion
Watchdog Timer	Independent and Window Watchdog	Independent and Window Watchdog	✅ Both supported

Low Power Features	✅ Yes (Ultra Low Power: STOP, STANDBY, etc.)	❌ No low-power modes	🔸 If battery/standby is needed, L432 is better
Interrupt Capability	EXTI lines on all GPIOs	EXTI lines on all GPIOs	✅ Both support interrupts for protection/fault handling
Peripherals	DAC, Op-Amp, Comparators	No DAC, No analog comparators	🔸 L432 better if you want analog functions
EMI/Analog Noise Immunity	✅ Good separation, designed for low noise	Moderate (depends on board layout)	🔸 L432 has better analog performance overall
Cost	₹350–450 (Nucleo or chip)	₹220–300 (Blue Pill–like boards or chip)	✅ F401 is noticeably cheaper
Dev Board	Nucleo-L432KC	Nucleo-F401RE / Custom board	✅ Both are easy to work with
Ease of Sourcing	Moderate	Very easy	✅ F401CCU6 is widely available in Indian markets
Temperature Range Variant (U3/U6)	U3: up to 85°C U6: up to 105°C	Commercial grade: up to 85°C	✅ Equivalent for most non-industrial projects
Power Consumption	✅ Ultra low power (~100 µA in STOP mode)	❌ Higher power consumption (mA range in idle)	🔸 Use L432 for battery or power-constrained systems

Using this table we can see that the **STM32F401** is easier to source,cheaper, has the same RAM and ROM.So this will be our final choice for the project.