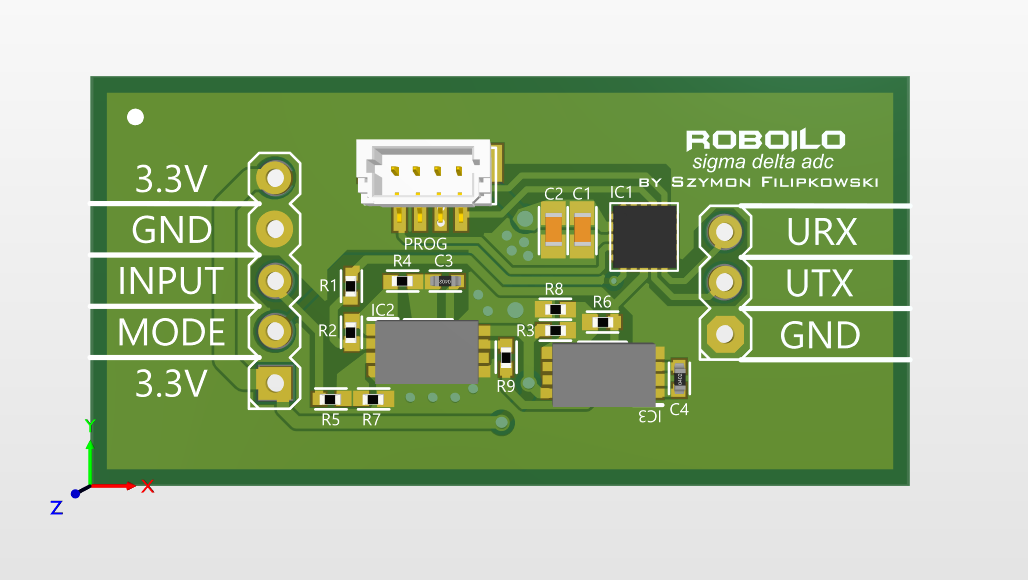
**SIGMA DELTA ADC MODULE**

**BY SZYMON FILIPKOWSKI**

Dokumentacja obejmuje Sigma Delta ADC module autorstwa Szymona Filipkowskiego, wersja 1.2.1. W niedalekiej przyszlosci projekt będzie dostępny na GITHUB: [“https://github.com/Tacot2009/Sigma-Delta-ADC-module-by-Szymon](https://github.com/Tacot2009/Sigma-Delta-ADC-module-by-Szymon-Filipkowski)”. Pliki zostaną dodane wkrótce. **Zużycie energii i częstotliwość próbkowania są w tej chwili szacunkami.**

**UWAGA!**

**Dokumentacja została pierwotnie napisana w języku angielskim, dlatego też zaleca się kożystać z niej, ponieważ może ona zawierać bardziej szczegółowe informacje niż te zawarte tutaj.**

****

**Spis treści**

1. Główne cechy
2. Przykładowe zastosowania
3. Opis
4. Historia wersji
5. Piny I/O i informacje o nich
6. Specyfikacja
7. Rekomendowane wartości pracy
8. Charakterystyka AC – tryb pojedyńczy
9. Charakterystyka AC – tryb ciągły
10. Dokładny opis
    1. Diagram
    2. Kontrola trybów pracy
    3. Opis działania
11. **Główne cechy**

* Wbudowany procesor
* Zakres wejściowy 0V do 2.56V
* Zasilanie 3.3V
* Komunikacja jednostronna UART
* Łatwa zmiana trybów pracy
* Funkcjonalność plug-and-play
* Najażniejsze cechy
  + Rozdzielczość: 8 Bitów
  + Dokładność 0.01V
  + Częstotliwość odczytów 2-10Hz
  + Zycie mocy u2mW

1. **Przykładowe zastosowania**

* Współpraca z dowolnym urządzeniem obsługującym protokół komunikacyjny UART
* Interfejs do czujników temperatury, źródeł napięcia, przetworników, fotorezystorów itp.

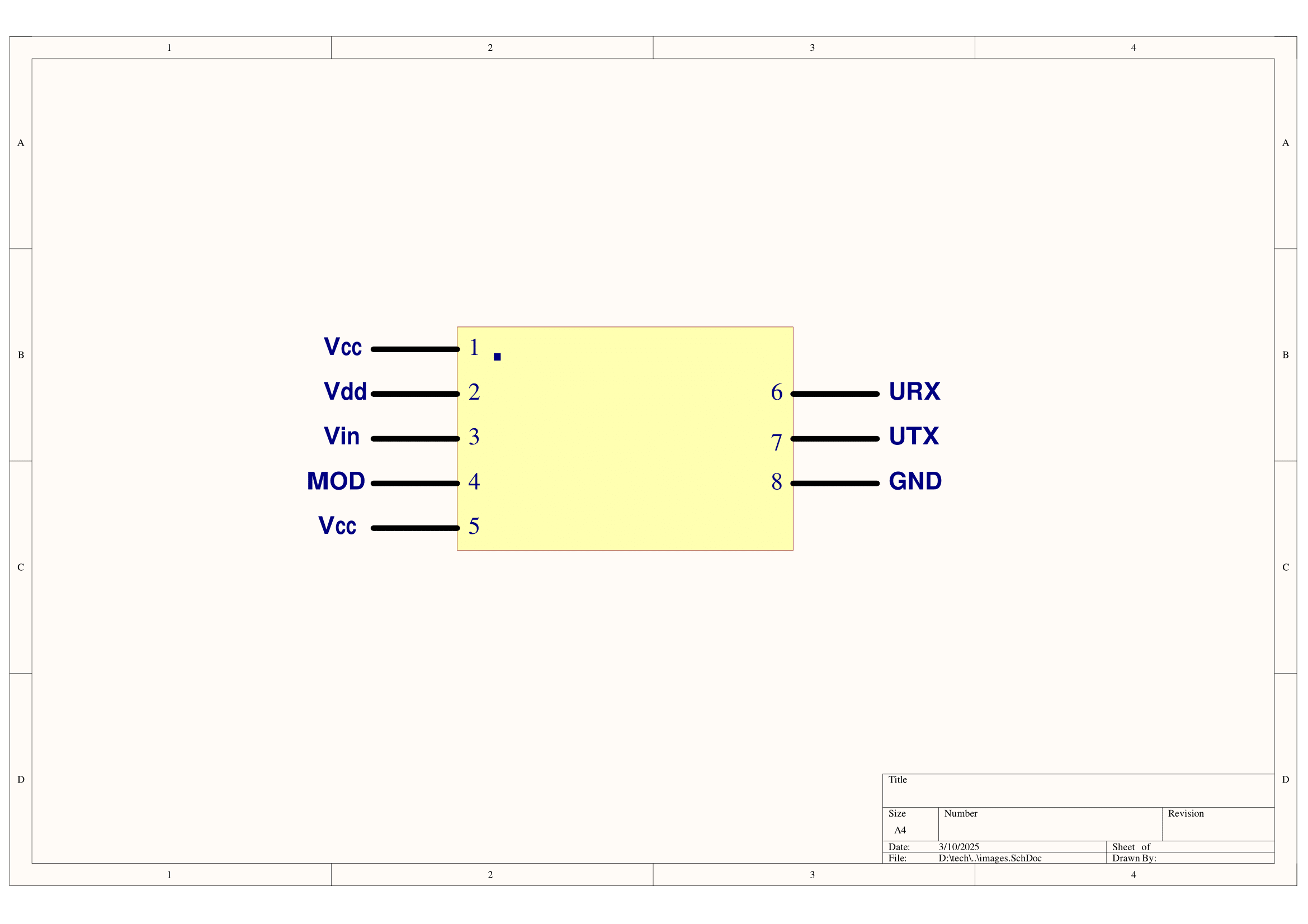
1. **Opis**

„Sigma Delta ADC module by Szymon Filipkowski” to 8-bitowy przetwornik ADC wykorzystujący modulację Sigma-Delta do szacowania wartości cyfrowej sygnału wejściowego. Moduł wykorzystuje mikrokontroler STM32C0 do dekodowania sygnału cyfrowego i przesyłania go w postaci wiadomości UART do urządzenia wyjściowego. Sprzęt i oprogramowanie są dostępne jako open-source na GITHUB, dzięki czemu każdy może dostosować ten moduł do własnych potrzeb. Autorem i twórcą projektu jest Szymon Filipkowski.

1. **Historia wersji**

* v1.0 - pierwszy projekt PCB, oprogramowanie w fazie rozwoju.
* v1.2.0 – dodanie wyjścia obsługującego komunikację UART z procesora do głównego złącza wyjść na PCB, przewidziana do przyszłych zastosowań.
* V.1.2.1 – dodanie dokumentacji w języku polskim
* v1.3.0 – dodano UART, dodano zlacze do programowania procesora

1. **Piny I/O i informacje o nich**

****

|  |  |  |  |
| --- | --- | --- | --- |
| **PIN** | | **I/O** | **DESCRIPTION** |
| **NO.** | **NAME** |
| 1. | Vcc | I | Zasilanie +3.3V |
| 2. | Vdd | I | Wejście masy (+0V) |
| 3. | Vin | I | Wejście napięcia analogowego |
| 4. | MOD | I | Wybór trybu pracy |
| 5. | CN1 | O | Wyjście napięcia Vcc |
| 6. | URX | I | Wejście danych protokołu UART |
| 7. | UTX | O | Wyjście danych protokołu UART |
| 8. | GND | O | Referencja 0V/masa dla komunikacji z procesorem |

1. **Specyfikacja**
   1. **Rekomendowane wartości pracy**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **MIN** | **MAX** | **UNIT** |
| Vcc | 3.3 | 3.3 | V |
| Vdd | 0 | 0 | V |
| Wejście napięcia analogowego | 0 | 2.56 | V |

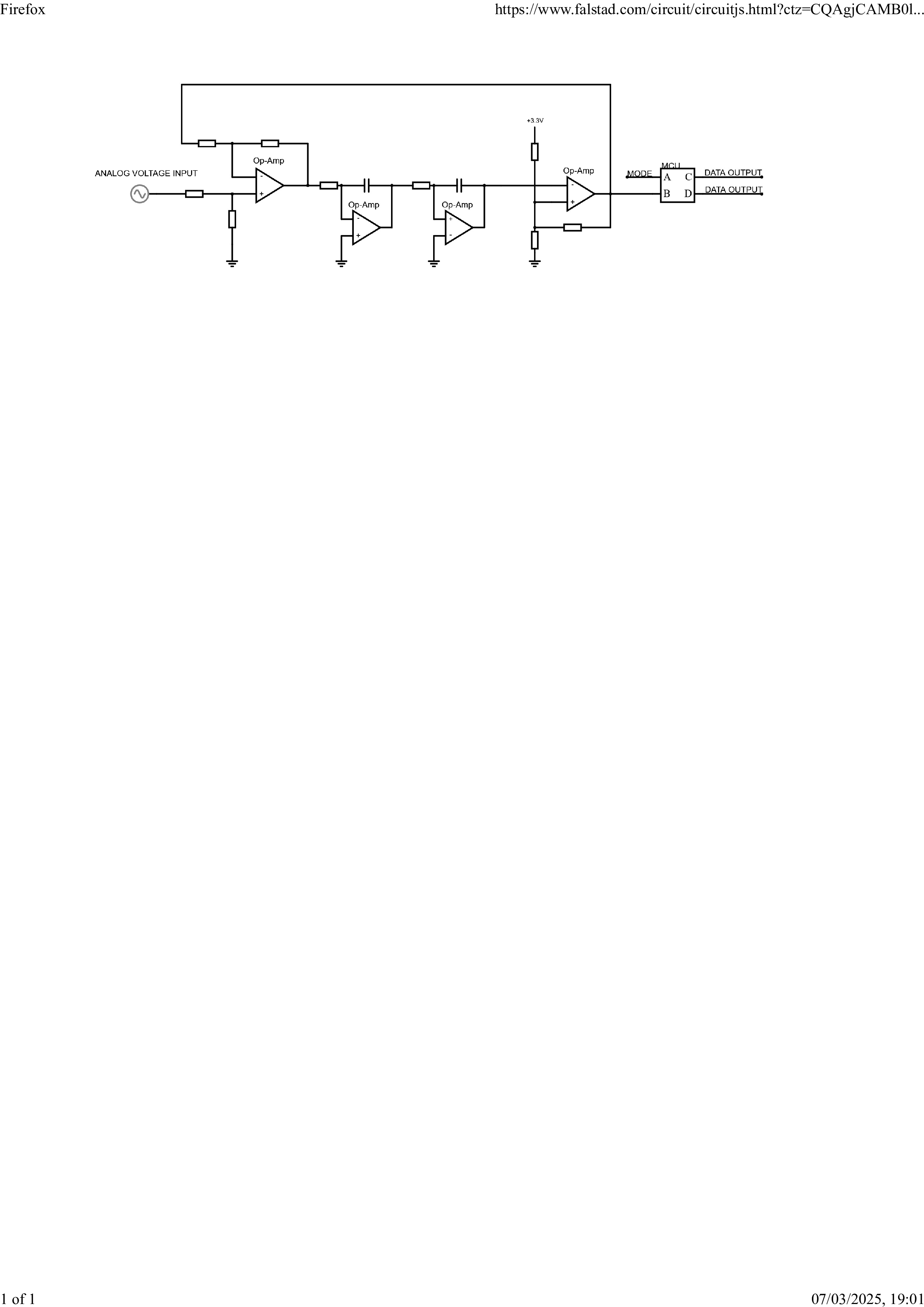
* 1. **Charakterystyka AC – tryb pojedy**

|  |  |  |
| --- | --- | --- |
|  | **TEST CONDITIONS** | **TYPICAL** |
| Czas pojedynczej konwersji | Vin = 3.3V | 1 SEC |
| Zużycie energii | Vin = 3.3V | 2 mW |

* 1. **Charakterystyka AC – tryb ciągły**

|  |  |  |
| --- | --- | --- |
|  | **TEST CONDITIONS** | **TYPICAL** |
| Częstotliwość konwersji | Vin = 3.3V | 2Hz |
| Zużycie energii | Vin = 3.3V | 2mW |

1. **Dokładny opis**
   1. **Diagram**



* 1. **Kontrola trybów pracy**

Urządzenie ADC obsługuje dwa tryby pracy: tryb wywołania (tryb pojedynczy) i tryb ciągły (tryb automatyczny). Tryb automatyczny jest domyślnym trybem modułu. Po zwarciu pinu MOD do 3.3V, moduł przechodzi w tryb pojedynczy. W tym trybie ADC wykonuje jedną konwersję napięcia i wysyła ją za pomocą protokołu UART. Następnie MCU przechodzi w tryb STOP, czekając na odłączenie pinu MOD od 3.3V. Pin MOD ma pull-down. Jeśli pin MOD pozostaje rozłączony, moduł działa w trybie automatycznym, przesyłając dane przez protokół UART po każdej zakończonej konwersji napięcia wejściowego.

* 1. **Opis działania**
     + 1. Sygnał analogowy jest pobierany bezpośrednio ze źródła.
       2. Wzmacniacz operacyjny w konfiguracji diffrence aplifier odejmuje napięcie sprzężenia zwrotnego (pochodzące z poprzedniego wyjścia) od sygnału wejściowego. Tworzy to sygnał błędu pokazujący różnicę między wejściem a aktualnym wyjściem.
       3. Dwa wzmacniacze operacyjne, działające jako integratory, przetwarzają sygnał błędu. Filtrują one szum o wysokiej częstotliwości i wprowadzają opóźnienie. To poprawia dokładność sygnału.
       4. Wzmacniacz operacyjny pełniący funkcję komparatora konwertuje zintegrowany sygnał na cyfrowe wyjście binarne. Powstałe wyjście cyfrowe służy zarówno jako wynik konwersji, jak i sprzężenie zwrotne dla ciągłej korekcji błędu.