

Dyanko K. Hubenov, Nikolay Zh. Kulev, Rosen A. Bogdanov,
ADALM-PLUTO BUILD SDR SYSTEM PROGRAMMING

Dyanko K. Hubenov, Nikolay Zh. Kulev, Rosen A. Bogdanov

*Communication Networks and Systems Department,
"Vasil Levski" National Military University, 1 "Karel Shkorpil" str., Shumen, Bulgaria*

e-mail: d_hubenov@abv.bg, nz_kulev@abv.bg, rab61@abv.bg

Abstract: *In this paper are given programming models for software reconfiguration of Adalm-Pluto build SDR system.*

Keywords: *SDR, Analog Devices, Zynq, Xilinx*

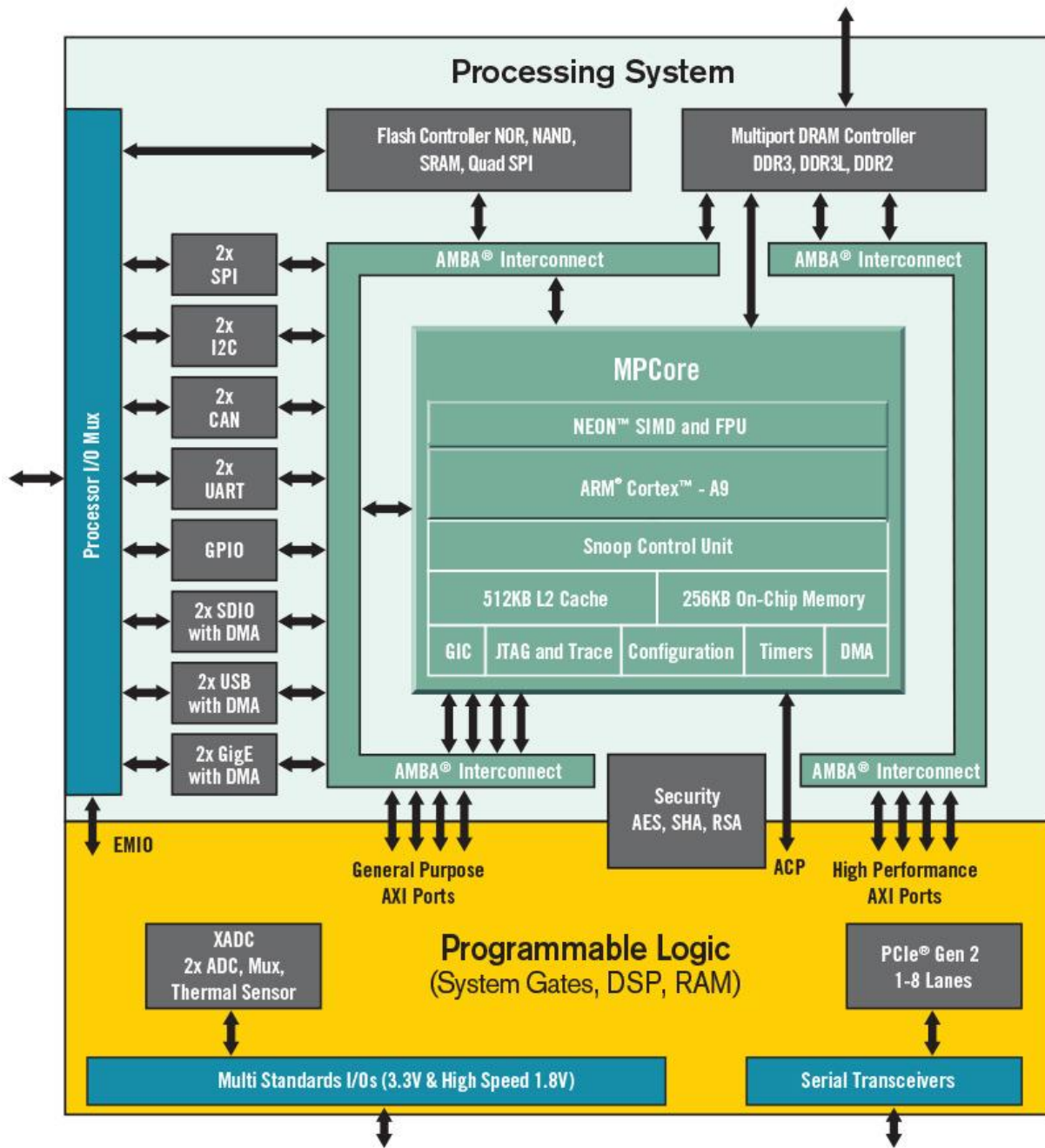
**ПРОГРАМИРАНЕ НА SDR СИСТЕМА, БАЗИРАНА НА
ADALM-PLUTO**

Дянко К. Хубенов, Николай Ж. Кулев, Росен А. Богданов

*Катедра „Комуникационни мрежи и системи, Факултет „Артилерия, ПВО и КИС“,
Национален военен университет „Васил Левски“, гр. Шумен, ул. „Карел Шкорпил“ 1*

Въведение

SDR системите изпълняват комбинация от множество задачи за обработка на данни, комуникация и потребителски интерфейс, които имат различни изисквания за ширина на лентата за обработка и ограничения при работа в реално време. Затова, хардуерната платформа, избрана за изпълнение на такава система трябва да е стабилна и мащабируема в същото време позволяваща бъдещи подобрения и разширения на системата. Xilinx Zynq-7000 All Programmable SoCs отговаря на тези изисквания чрез предоставяне на високопроизводителна система за обработка, комбинирана с програмируема логика, както е показано на фигура 1. Комбинацията от програмируема логика и система за обработка осигурява изключителна паралелна мощност на обработка, ефективност в реално време, бързи изчислителни скорости и гъвкавост на свързаността.



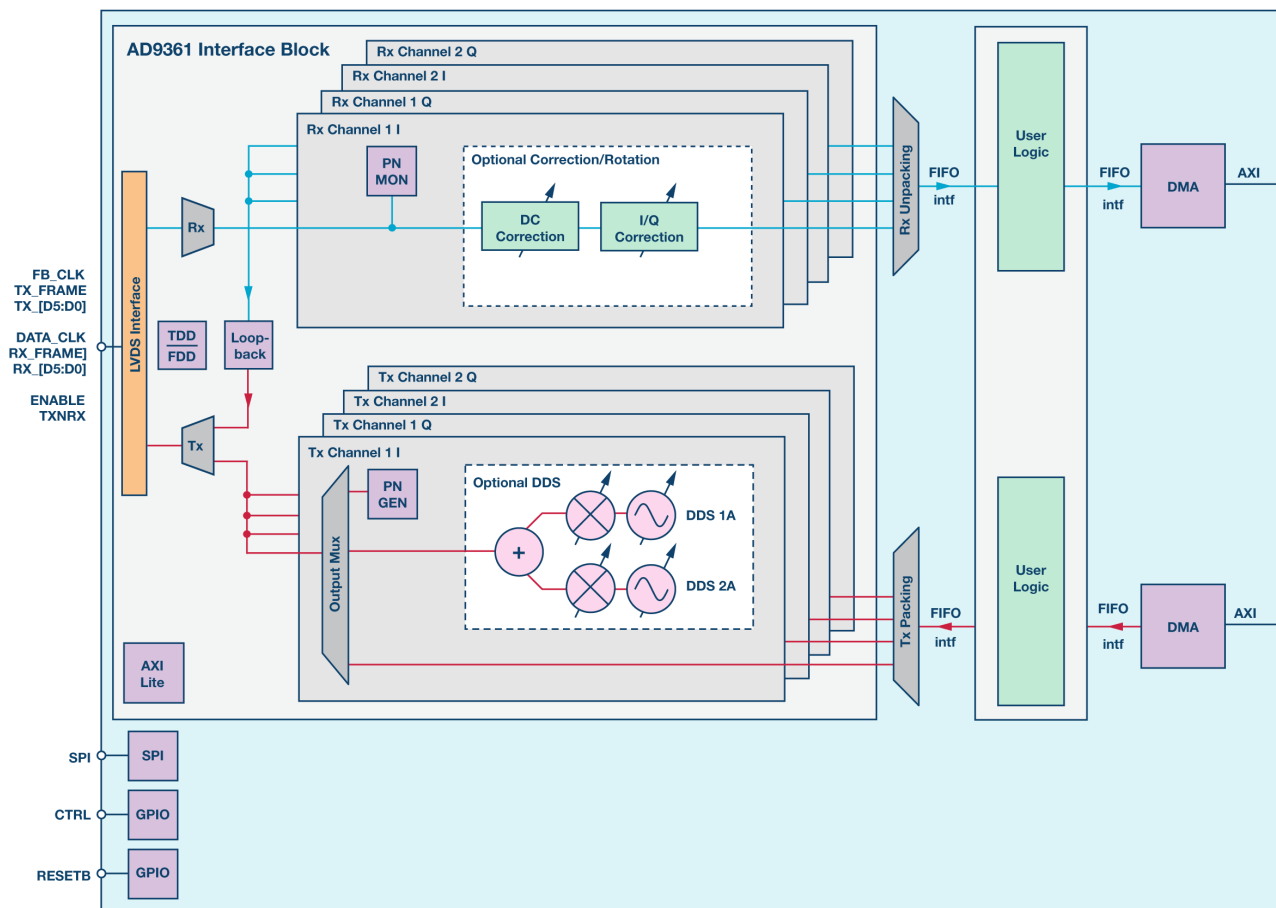
Фигура. 1. Функционална блокова схема на Xilinx Zynq SoC [3]

1. Софтуерна инфраструктура

Програмируемата система Xilinx Zynq All Programmable System on Chip (AP SoC) интегрира софтуерната програмируемост на ARM-базиран процесор с хардуерната програмируемост на FPGA, което позволява хардуерно ускоряване при интегриране на CPU, DSP, стандартен продукт със специфично приложение ASSP (Application-Specific Standard Product) и смесена сигнална функционалност в едно устройство. Устройството разполага с двудрен процесор ARM Cortex™ -A9, който е свързан с 28 nm програмируема логика, базирана на Artin® 7. Оборудван е и с най-често използваните хардуерни периферни устройства като USB и SPI.

Фигура 2 показва инфраструктурата на Analog Devices Zynq за поддръжка на ADALM-PLUTO и платките FMCOMMSx с възможности за използване на hardware description language HDL и софтуерен дизайн.

Всички ADI Linux драйвери са базирани на Linux индустриална входно/изходна подсистема Linux IO, която е включена във всички основни Linux ядра. IO Score е приложение с отворен код за Linux, разработено от Analog Devices [2], което работи на двудрения ARM Cortex-A9 процесор в Xilinx Zynq и има способността да представя данни в реално време, придобити от всички аналогови устройства с FPGA Mezzanine Card (FMC) карта, свързани към платформата Xilinx Zynq. Данните може да се представят както във времева и честотна област, така и като констелация.



Фигура 2. ADI HDL и софтуерна инфраструктура [4]

2. Програмиране на SDR системата

За тестване на софтуерните възможности за пренастройване на когнитивната радиосистема, базирана на хардуерната платформа ADALM-PLUTO, са използвани възможностите за програмиране на вграденото FPGA, което предоставя няколко варианта за генериране на предаваните данни. Първата възможност използва вграденият директен цифров синтезатор DDS (Direct Digital Synthesizer) за предаване на едно- и дву-тонален сигнал по канали I и Q на DAC. Втората възможност използва функцията за директен достъп до паметта DMA (Direct Memory Access) за предаване на персонализирани данни, които са съхранени във файл [1].

Управлението на приемопредавателя [5,6] в PlutoSDR се осъществява чрез IO подсистемата на Linux. Драйверите на устройството позволяват да се вземат проби чрез IO подсистемата на Linux ядрото, управляваща функциите за аналоговоцифрови или цифровоаналогови преобразуватели и свързаните с тях хардуер. IO комуникира с потребителското пространство чрез sysfs и символни устройства чрез извикване на функция или метод. Библиотеката LibIO осигурява всякакъв вид абстракция на устройствата и обработка всички видове вътрешни IO.

➤ Управление на приемопредавателя

Управлението на приемопредавателя се извършва в четири стъпки:
Създаване на ПО IP Network, зададена чрез IP адрес ip:xxx.xxx.xxx.xxx или usb:XX.XX.X
Получаване на структурата на физическия слой на AD936x PHY device
Задаване на честотата на TX LO
Задаване на честотата на дискретизация RX
Програма на C++ за управление на предавателя е представена на Листинг 1.

Листинг 1. Управление на приемопредавателя

```
#include <iio.h>

int main (int argc, char **argv)
{
    struct iio_context *ctx;
    struct iio_device *phy;

    ctx = iio_create_context_from_uri("ip:192.168.2.1");

    phy = iio_context_find_device(ctx, "ad9361-phy");

    iio_channel_attr_write_longlong(
        true,
        iio_device_find_channel(phy, "altvoltage0",
            "frequency",
            2400000000); /* TX LO честота = 2.4GHz */

    iio_channel_attr_write_longlong(
        false,
        iio_device_find_channel(phy, "voltage0",
            "sampling_frequency",
            5000000); /* RX скорост = 5 MSPS */

    receive(ctx);

    iio_context_destroy(ctx);

    return 0;
}
```

➤ Управление на приемането на данни

Управлението на приемането на данни се извършва в 6 стъпки:
Получаване на RX структурата на устройството;
Получаване на входните IQ канали;
Разрешаване на I и Q каналите;
Създаване на RX буфер;
Запълване на буфера;

1. Обработка на пробите.

Програма на C++ за управление на приемането на данни е представена на Листинг 2.

Листинг 2. Управление на приемането на данни

```
int receive(struct iio_context *ctx)
{
    struct iio_device *dev;
    struct iio_channel *rx0_i, *rx0_q;
    struct iio_buffer *rxbuf;

    dev = iio_context_find_device(ctx, "cf-ad9361-lpc");

    rx0_i = iio_device_find_channel(dev, "voltage0", 0);
    rx0_q = iio_device_find_channel(dev, "voltage1", 0);

    iio_channel_enable(rx0_i);
    iio_channel_enable(rx0_q);

    rxbuf = iio_device_create_buffer(dev, 4096, false);
    if (!rxbuf) {
        perror ("Could not create RX buffer");
        shutdown();
    }

    while (true) {
        void *p_dat, *p_end, *t_dat;
        ptrdiff_t p_inc;

        iio_buffer_refill(rxbuf);

        p_inc = iio_buffer_step(rxbuf);
        p_end = iio_buffer_end(rxbuf);

        for (p_dat = iio_buffer_first(rxbuf, rx0_i);
p_dat < p_end; p_dat += p_inc, t_dat += p_inc) {
            const int16_t i = ((int16_t*)p_dat)[0];
//Real(I)
            const int16_t q = ((int16_t*)p_dat)[1];
//Imag(Q)

                /* Обработка на пробите */
            }
        }
        iio_buffer_destroy(rxbuf);
    }
}
```

Програмируемата хардуерна архитектура на когнитивна радиосистема, базирана на ADALM-PLUTO, позволява да се изпълняват множество задачи в реално време, като обработка на данни, комуникация и потребителски интерфейс. Комбинацията от програмируеми RF интерфейс, логика и система за обработка предоставя възможност за паралелна обработка, бързи изчислителни процеси, ефективност при работа в реално време, и гъвкавост на свързаността. Това от своя страна, осигурява възможност за бъдещи подобрения и разширения на когнитивната радиосистема.

References

1. Analog Devices, ADALM-PLUTO: SDR Active Learning Module, 2017, Available at <http://www.analog.com/media/en/news-marketing-collateral/product-highlight/ADALM-PLUTO-Product-Highlight.pdf>.
2. Analog Devices, RF Agile Transceiver AD9363, 2016, Available at <http://www.analog.com/media/en/technical-documentation/data-sheets/ad9363.pdf>.
3. Xilinx, Zynq-7000 SoC Data Sheet: Overview, 2018. Available at https://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf.
4. Xilinx, Zynq-7000 SoC Technical Reference Manual, 2018. Available at https://www.xilinx.com/support/documentation/user_guides/ug585-Zynq-7000-TRM.pdf.
5. Димитров, Д. И., Анализ на информационния потенциал на радиотехнически системи, използващи ефекта на нелинейно отражение на електромагнитните вълни, Научни трудове, Русенски университет „А. Кънчев“, Том 47, Серия 3.2, с. 68-72, 2008, ISSN 1311-3321.
6. Димитров, Д. И., Василев, В. М., Особенности на параметрите на сондиращия сигнал в нелинейна радиолокационная станция, „Военно издателство“, София, с. 36-50, 2008, ISSN 1312-2916.