

Laborator 7 TPI 2022-2023

Prelucrarea secvențelor video folosind placa de dezvoltare

EZ-KIT Lite Blackfin BF 533

1. Achiziția semnalelor video pe placa de dezvoltare EZ-KIT lite

[Placa de dezvoltare EZ-KIT lite](#) permite dezvoltarea de aplicații audio și video pentru prelucrarea semnalelor pe procesoare Blackfin.

Pentru procesarea semnalelor video placa de dezvoltare include codec-uri dedicate: [ADV7171](#) (codor video) și [ADV7183](#) (decoder video) pentru redare pe un monitor extern sau transmisie sau, respectiv, pentru conversie în format digital.

În mod captură de semnal video interfațarea hardware între sursa de semnal video și decoderul ADV7183 se efectuează prin intermediul interfeței **PPI (Parallel Peripheral Interface)**.

Visual DSP++ include proiecte de tip demo atât pentru achiziție cât și pentru redare de secvențe video.

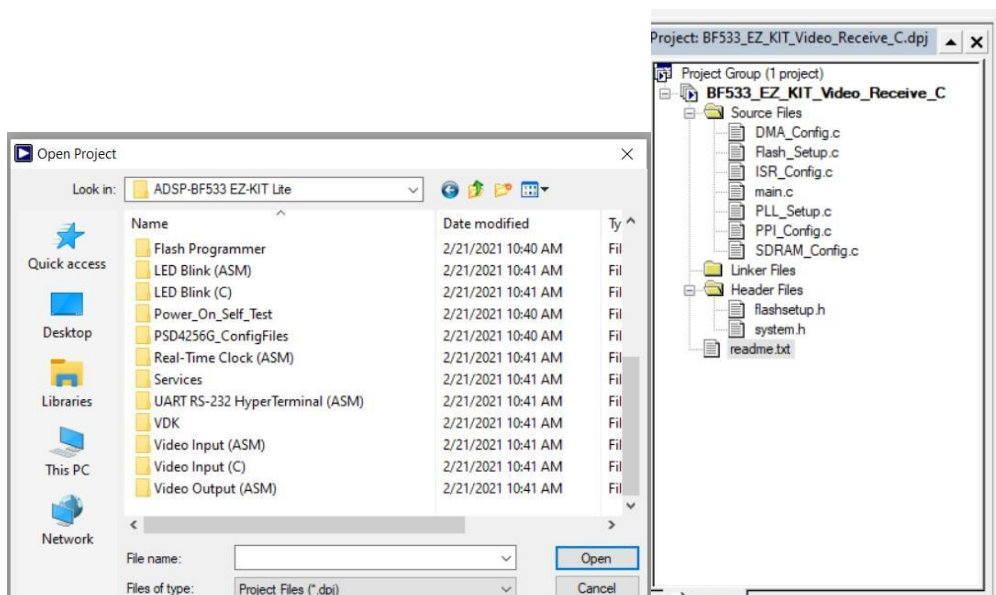


Fig.1 Proiecte demo EZ-KIT Lite. Structura proiectului pentru achiziție de semnale video.

Aplicația din Fig. 1 poate fi configurată pentru achiziția de semnale conform specificației [ITUR 656-3](#) și include configurarea în registrelor PPI dedicate mapate în memorie. Conectarea propriu-zisă a unei surse de semnal video se face prin intermediul unor [conectori dedicați](#) iar transferul cadrelor video în memoria SDRAM se face via controllerul DMA la o adresă predefinită în fișierele de configurare.

Aplicația poate fi configurată pentru achiziția de semnale video cu scanare progresivă sau cu **scanare întrețesută** (fiecare cadru este împărțit în câmpuri conținând linii impare și, respectiv pare).

Rularea aplicației permite achiziția unei imagini care poate fi vizualizată cu ajutorul utilitarului *Image Viewer*. Figura 2 prezintă un rezultat obținut prin rularea aplicației în mod scanare întrețesută și pune în evidență liniile pare și respectiv impare care formează un cadru.

Reprezentarea în memorie este de forma:

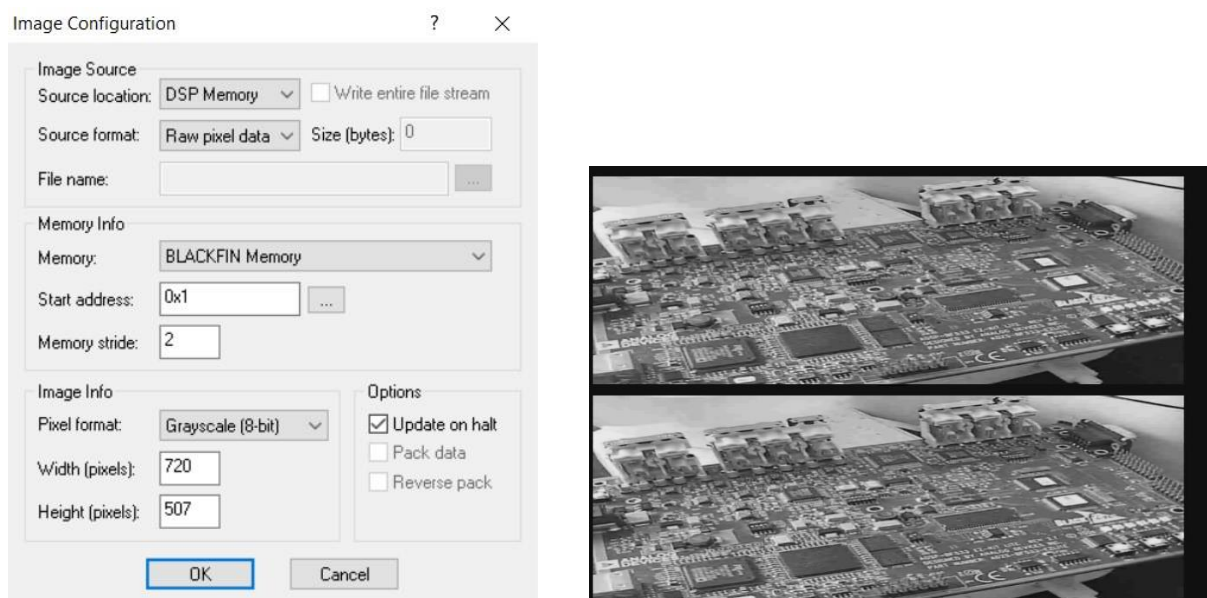
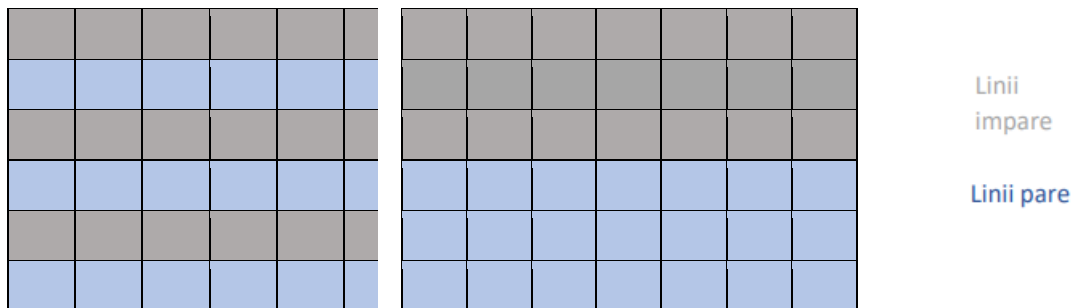


Fig.2 Setări *Image viewer*. Liniile impare și respectiv pare stocate în memorie

Sursa de intrare pentru figura 2 a fost un semnal video NTSC. În acest semnal fiecare câmp activ are 720 x 507 pixeli, fiecare pixel folosind doi octeți pentru stocarea informației, un octet pentru informația de luminanță și un octet pentru informația de culoare. Setarea parametrului **memory stride** la valoarea 2 (octeți) permite citirea doar a informațiilor de luminanță. Utilitarul permite în acest fel vizualizarea informației în formatul în care aceasta a fost capturată (scanare întrețesută) iar pentru dezvoltarea unei aplicații de procesare a informației din semnale video capturate în format digital trebuie să se țină de această modalitate de reprezentare în memorie.

2. Reprezentarea informației video în memorie. Deîntrețeserea câmpurilor pare și impare

Pentru prelucrarea unei imagini incluse într-un cadru video imaginea stocată în memorie sau pe hard-disk poate fi mai întâi reprezentată în formatul original. Secvența de cod C din figura 3 permite refacerea imaginii de intrare prin deîntrețesere.

Pentru imaginea de ieșire din Fig.2, stocată pe hard-disk și încărcată pe placa de dezvoltare în tabloul *in*, rezultatul rulării aplicației este ilustrat în Fig. 3.

```
#define N 720
#define M 507

unsigned char in[N*M];
unsigned char out[N*M];

void main ()
{
    int linie;
    for (linie=0;linie<=M/2;linie+=1)
        copiere_linie (&in[linie*N], &out[linie*2*N],N);
    for (linie=M/2+1;linie<=M;linie+=1)
        copiere_linie (&in[linie*N], &out[N*((linie-M/2)*2-1)].N);
}

void copiere_linie (unsigned char *src, unsigned char *dest, int size)
{
    int i;
    for (i=0;i<size;i++)
        dest[i]=src[i];
}
```

Fig.3 Funcție main pentru un proiect C care permite deținerea imaginilor stocate pe hard-disk.

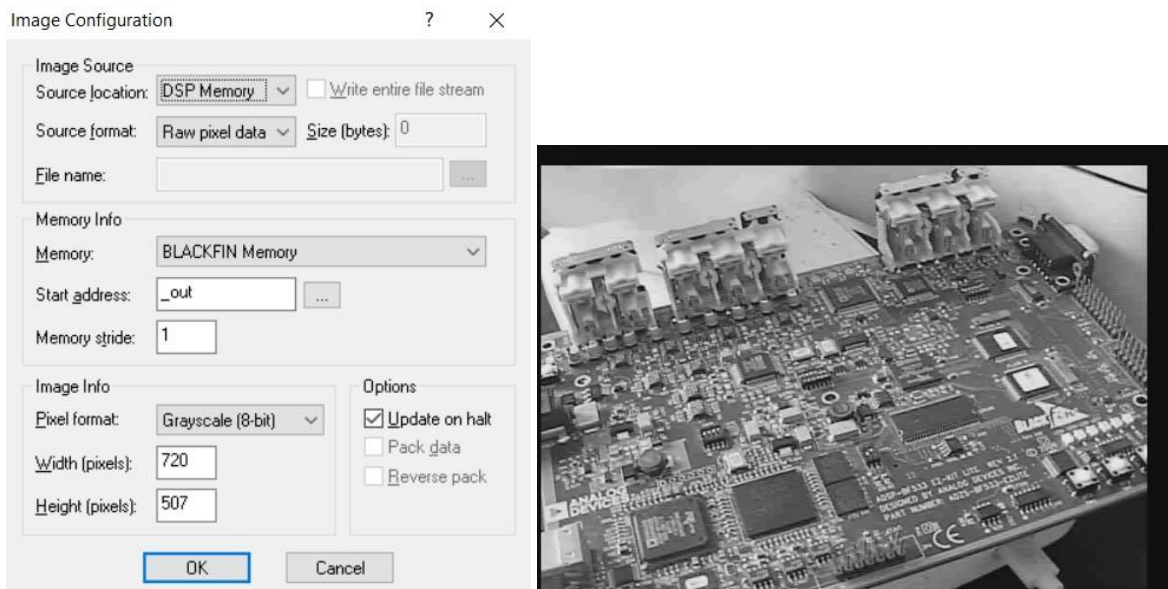


Fig.4 Rezultat deținere pentru o imagine capturată în format NTSC întretesut

3. Exerciții

- 1) Analizați și explicați modul de implementare a operației de deținere
- 2) Integrați operația de deținere în aplicația demo furnizată de Analog Devices pentru captura de semnale video.
- 3) Propuneti o implementare pentru un filtru trece jos de tip medie de dimensiuni 3x3 pixeli și, respectiv, un filtru wavelet VisuShrink care să opereze pe cadrul video capturat, deținut.
- 4) Vizualizați rezultatul folosind utilitarul *Image Viewer* setat corespunzător.

5) Propuneti o implementare pentru filtrele de la punctul 3 care să țină cont de modalitatea de reprezentare în memorie a unui cadru video NTSC și care să nu folosească funcții de deîntreșere.