

## Aplicația III.5

# PRELUCRAREA ȘI GENERAREA SEMNALELOR NUMERICE

### 5.1. Scopul lucrării

În aplicația III.4 s-a făcut precizarea că, în multe cazuri din lumea reală, semnalele sunt sub formă numerică (digitală). Automatizările clasice și moderne folosesc frecvent dispozitive de acționare la care comanda este de tip binar. De exemplu, într-o instalație automatizată de încălzire se impune menținerea temperaturii între două limite, ceea ce este echivalent cu a porni elementul încălzitor atunci când temperatura se află sub limita inferioară, respectiv a-l opri atunci când se atinge limita superioară.

De asemenea, în cadrul automatizărilor numerice, se folosesc elemente acționate binar, cum ar fi relele, contactoarele intermediare și de forță etc.

Firmele producătoare de sisteme de achiziție / distribuție a datelor au oferte variate privind generarea comenzilor digitale, care țin seama de rata de înprospătare a ieșirilor digitale, numărul canalelor digitale prezente simultan în ieșire, natura sarcinii pe care poate “debita” semnalul sistemul de distribuție respectiv etc. Pentru mai multe detalii privind astfel de sisteme – fie destinate numai achiziției de semnale analogice/numerice – fie numai pentru distribuția de comenzi analogice/numerice – fie sisteme multifuncționale înglobând toate funcțiile de mai sus, se pot studia ofertele unor firme de referință în acest domeniu.

Scopul lucrării este de a genera unui mediu extern calculatorului comenzi digitale de natura unor tensiuni compatibile logicii TTL, folosind facilitățile de prelucrare și afișare ale mediului de dezvoltare **LabVIEW**. În același timp, această lucrare se constituie ca o demonstrație a modului de elaborare a comenzilor numerice, fără a avea pretenția de a “pune” o instalație la lucru, așa cum se pot urmări exemplificările din aplicațiile III.3 și III.6.

### 5.2. Chestiuni de studiat

2.1. Se vor genera tensiuni digitale TTL, care vor fi puse să comande dispozitivul de afișare numerică cu 7 segmente **SN 7545**.

2.2. Se va simula o structură de numărător binar pe 4 ranguri, un decodificator pentru 7 segmente și un afișor pe 7 segmente, care va fi pusă să comande dispozitivul de afișare numerică cu 7 segmente **SN 7545**.

### 5.3. Schema de montaj și aparatura utilizată

Pentru generarea tensiunilor digitale TTL, care vor fi puse să comande dispozitivul de afișare numerică cu 7 segmente se va realiza montajul din fig.5.1, în care:

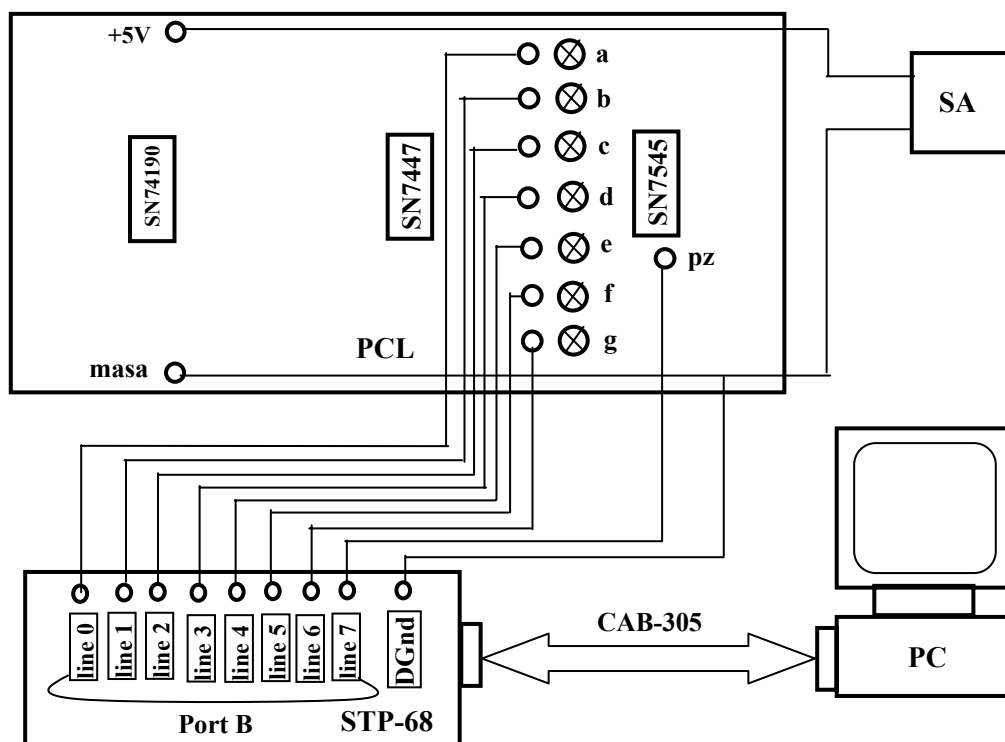


Fig.5.1. Schema montajului pentru generarea semnalelor digitale TTL cu placa tip plug-in **KPCI-3102**

**PCL** – placă cu circuite logice, pe care sunt montate numărătorul, decodificatorul pentru 7 segmente și afişajul cu 7 segmente;

**SA** – sursă de alimentare de 5V și 1A;

**STP-68** – placă de conexiuni externe (prin înfiletare), legată prin intermediul cablului 1:1 **CAB-305** la conectorul din spatele **PC**-ului aferent plăcii **KPCI-3102**;

**PC** – calculator personal cu software-ul instalat pentru realizarea aplicației de generare a semnalelor digitale.

### 5.4. Panoul frontal și diagrama aplicației

În cadrul lucrării se folosește placa multifuncțională de intrări-ieșiri analogice și numerice **KPCI-3102** de fabricație KEITHLEY, de tip plug-in, compatibilă cu magistrala de **PC**, cu specificațiile (performanțele) și conexiunile externe prezentate în *ANEXA G*.

Întrucât dezvoltarea aplicației se face în **LabVIEW**, placa **KPCI-3102** trebuie “instalată” pe calculator, în acest scop efectuându-se **pașii 1...4** expuși în secțiunea **Instalare, configurare, montaje tipice** din *ANEXA G*. Se consideră că s-au efectuat pașii anterior precizați, așadar placa este pregătită – la modul general – pentru dezvoltarea aplicațiilor în **LabVIEW**. Totuși, înainte de dezvoltarea aplicației, se lansează din **Start >> Programs >> DriverLINX >> DriverLINX Configuration Panel**, se selectează **Device 0**, după care – cu opțiunea **Configure** – se selectează **DIO Ports**, configurând **portul 0** ca intrări digitale, iar **portul 1** ca ieșiri digitale.

În dezvoltarea aplicației conform cerințelor de la chestiuni de studiat, se pleacă de la ipoteza că pe **PC-ul** în care este introdusă placa **KPCI-3102** s-au respectat pașii de instalare software specificați în *ANEXA G*, ca și configurarea celor două porturi așa cum s-a precizat anterior.

Se prezintă – în continuare – configurația diagramei și a panoului frontal pentru situația generării semnalelor digitale către dispozitivul de afișare cu 7 segmente **SN 7545** conform cerinței 2.1 de la chestiuni de studiat, pe **portul B (1)** al plăcii **KPCI-3102**.

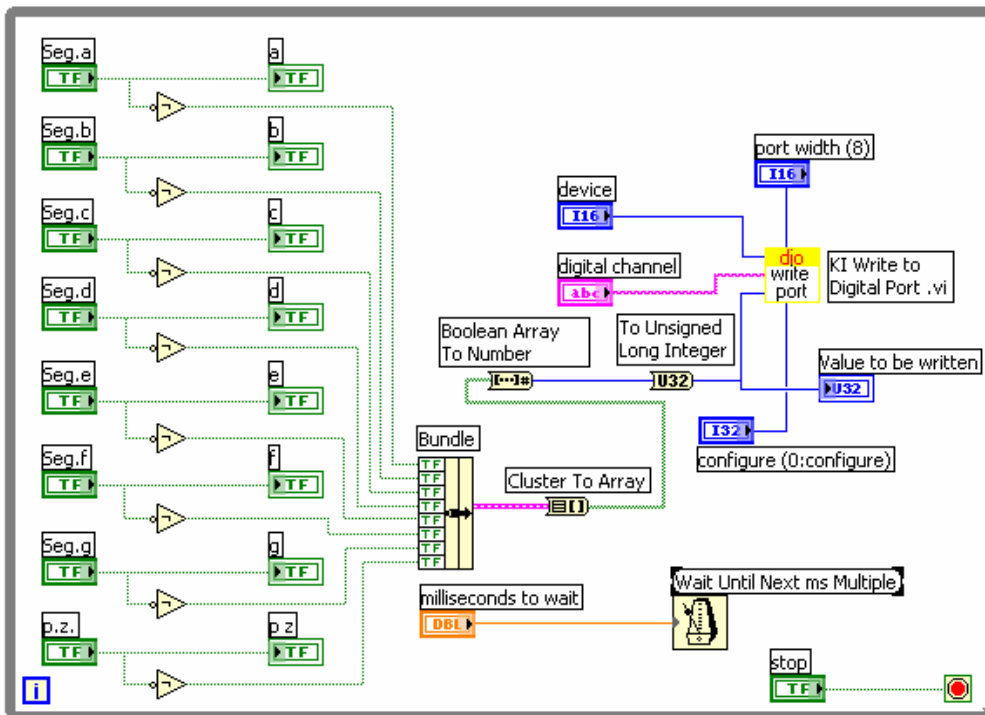


Fig.5.2. Diagrama bloc a aplicației de generare a comenzilor digitale pe **portul B(1)** al plăcii plug-in **KPCI-3102**

În fig.5.2. este prezentată diagrama bloc, în timp ce fig.5.3 arată configurația panoului frontal a aplicației pentru generarea semnalelor digitale conform primului punct al chestiunilor de studiat.

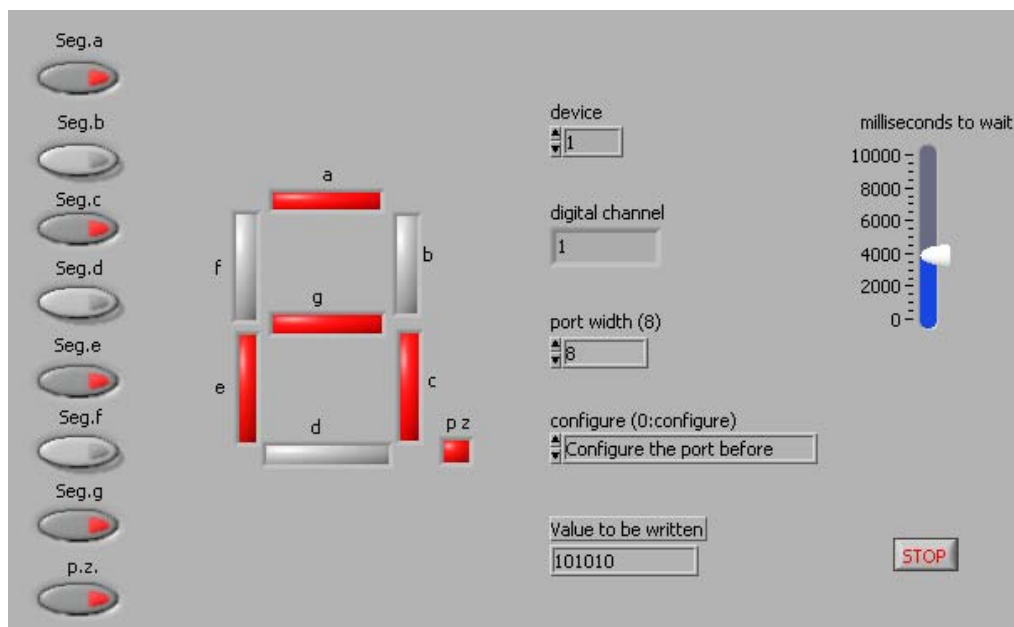


Fig.5.3. Panoul frontal pentru aplicația de generare comenzi digitale prin portul **B(1)** al plăcii plug-in **KPCI-3102**

Pentru construcția rapidă a panoului frontal, respectiv a diagramei, se recomandă realizarea etapelor, după cum urmează:

- se deschide un nou **VI**, în panoul frontal trăgându-se din paleta **Functions** >> **Boolean** elementele: opt **Push Button** – cărora li se atribuie etichetele **Seg.a**, **Seg.b**, ..., **Seg.g**, **p.z.**, respectiv opt **Round LED**-uri, cărora li se modifică forma conform panoului frontal din fig.5.3 și li se atribuie etichetele **a**, **b**, ..., **g**, **p.z.** Se aranjează butoanele și LED-urile astfel încât să arate ca în fig.5.3. Segmentele afișorului simulat în aplicație, ca și butoanele, pot fi colorate pentru cele două stări logice folosind unealta **Set Color** din paleta **Tools** (așa se explică culoarea atribuită celor două categorii de obiecte din fig.5.3).

- din paleta **Functions** >> **Numeric** se aduc în panoul frontal elementele: trei **Digital Control** – cărora li se atribuie etichetele **device**, **configure(0:configure)**, **port width (8)** – un **Vertical Pointer Slide** la care eticheta se schimbă în **milliseconds to wait** și un **Digital Indicator** cu eticheta **Value to be written**; la indicatorul cu eticheta **Value to be written** se adaugă eticheta **Radix**, iar din opțiunea **Format & Precision** se alege **Binary** (astfel configurat, indicatorul va afișa starea curentă a octetului generat în biți);

- din paleta **Controls** >> **String & Path** se trage în panou un **String Control**, căruia i se atribuie eticheta **digital channel**;

- pentru a încheia definitivarea panoului frontal, din **Controls** >> **Boolean** se trage un **Stop Button**, care va fi folosit pentru oprirea aplicației; se aranjează

elementele de pe panoul frontal, astfel încât – pe cât posibil – imaginea acestuia să corespundă fig.5.3;

- se trece în diagrama bloc, constatându-se poziționarea elementelor construite în panoul frontal diferită de imaginea din fig.5.2; se aranjează corespunzător elementele respective – conform fig.5.2 – având în vedere că urmează a fi introduse alte obiecte suplimentare din paleta **Functions**; aplicația fiind doar în stadiul incipient, săgeata **Run** plasată pe bara de comenzi va fi întreruptă;

- se trag în diagramă obiectele lipsă, după cum urmează:

- din paleta **Functions** >> **User Libraries** >> **Keithley DAQ VIs** >> **KI Easy VIs** se trage în diagramă un **KI Write To Digital Port .vi**;
- din paleta **Functions** >> **Cluster** se aduce un **Bundle**, căruia i se mărește dimensiunea la **8**, respectiv un **Cluster To Array**;
- din paleta **Functions** >> **Numeric** >> **Conversion** se aduc în diagramă elementele **Boolean Array To Number** și **To Unsigned Long Integer**, iar din paleta **Functions** >> **Time & Dialog** se aduce un **Wait Until Next ms Multiple**;
- din paleta **Functions** >> **Boolean** se aduc în diagramă opt inversoare **Not**, care au rolul de a nega fiecare comandă digitală pentru segmentele și punctul zecimal al afișorului de pe placa externă (cerință impusă de modul de lucru al afișorului plasat pe **PCL**).

În acest moment, toate elementele din diagrama bloc arătată în fig.5.2 sunt prezente, trecându-se la conectarea acestora conform figurii. La încheierea tuturor conexiunilor, se construiește o buclă **While** activând **While Loop** din paleta **Functions** >> **Structures**, căreia i se alege opțiunea **Stop If True**. Prin încheierea acestei ultime operații, săgeata **Run** de pe bara de comenzi devine continuă, semnificând corectitudinea realizării întregii aplicații. Se salvează aplicația, în directorul aferent grupei, sub denumirea **generare comenzi afisor\_[nume/prenume student]\_[grupa].vi**.

Pentru realizarea aplicației conform cerinței 2.2 de la chestiuni de studiat, se realizează diagrama bloc din fig.5.4, cu panoul frontal reprezentat în fig.5.5.

La o analiză atentă a diagramei din fig.5.4 se constată că aceasta cuprinde o parte de generare construită pe baza unor primitive din **LabVIEW**, un numărator binar pe 4 ranguri și un decodificator pentru 7 segmente, care sunt **subVI**-uri existente în directorul **My Documents**, și o parte de comandă externă prin portul **B(1)** similară diagramei din fig.5.2. În consecință, pentru realizarea rapidă a aplicației, se vor efectua pașii, după cum urmează:

- cu aplicația anterioară oprită, se deschide un nou **VI**, în care – prin procedee **Copy/Paste** – se copiază în diagramă aplicația **generare comenzi afisor\_[nume/prenume student]\_[grupa].vi**;

- se mărește bucla **While** corespunzător spre stânga, având în vedere că urmează a se introduce obiecte suplimentare;

- se trece în panoul frontal din care se șterg – prin selectare cu săgeata din paleta **Tools** – controalele aferente **Seg.a**, **Seg.b**, ..., **Seg.g**, **p.z.**;

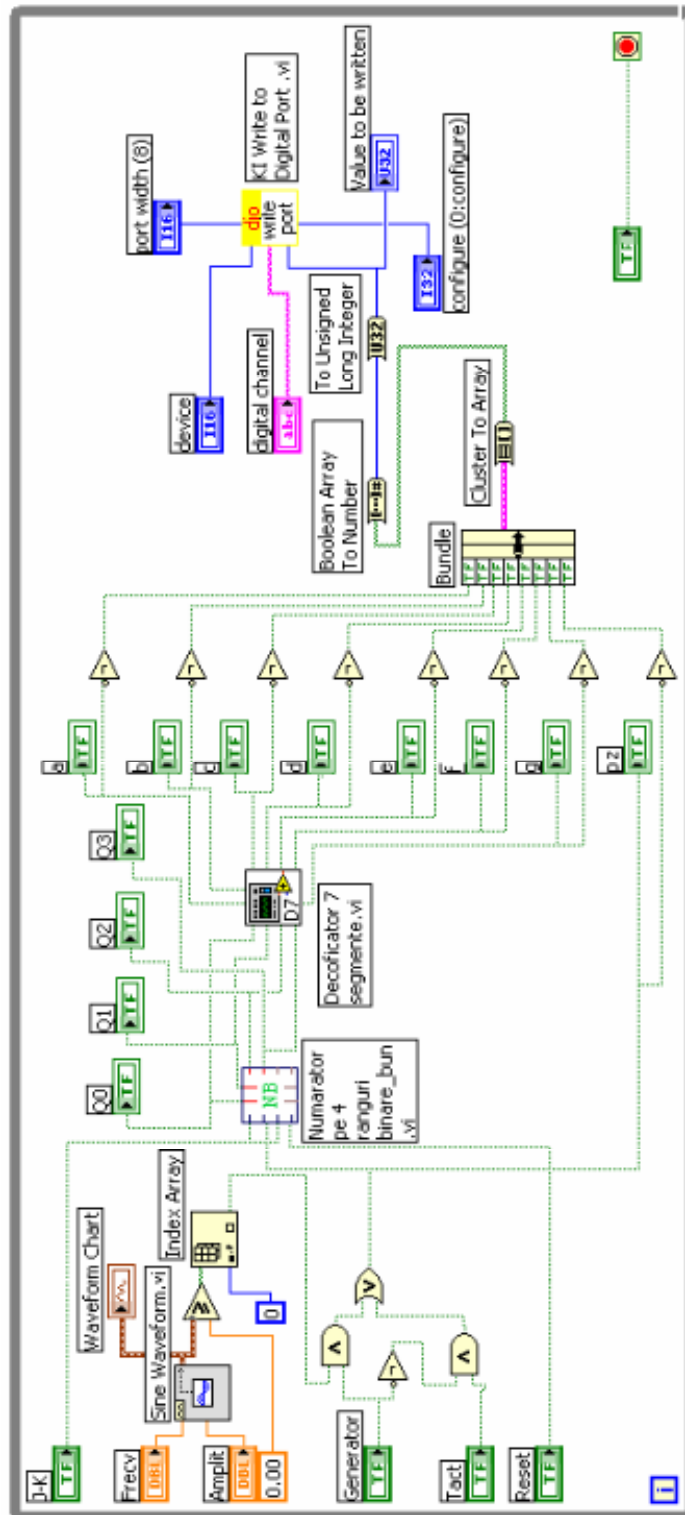


Fig.5.4. Diagrama bloc a aplicației de simulare structură de numărare, decodificare, afișare locală și comanda dispozitivului de afișare numerică SN 7545 prin intermediul plăcii KPCI-3102

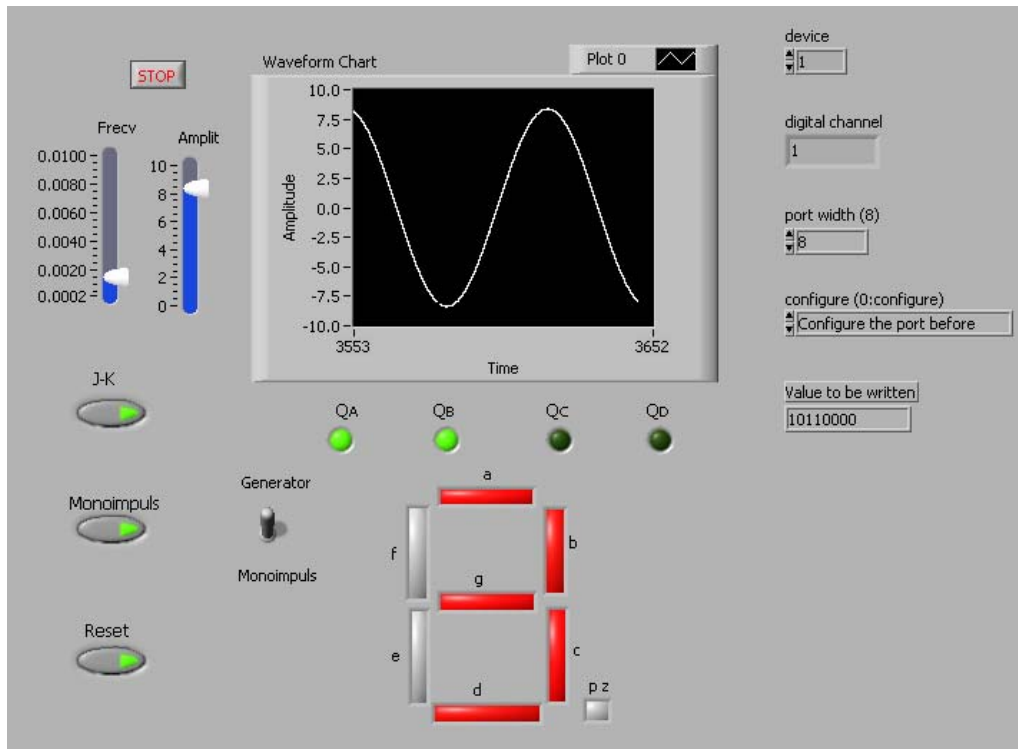


Fig.5.5. Panoul frontal al aplicației de simulare structură numărător, decodificator, afișor și comandă dispozitiv de afișare numerică cu 7 segmente SN 7545 folosind placa plug-in KPCI-3102

- din paleta **Controls** >> **Numeric** se trag în panoul frontal două controale **Vertical Pointer Slide**, cărora li se atribuie etichetele **Frecv**, respectiv **Amplit**, iar forma de reprezentare și gradarea fiecăruia se face în concordanță cu reprezentarea din fig.5.5;

- din paleta **Controls** >> **Graph** se aduce în panou un **Waveform Chart**, iar din paleta **Controls** >> **Boolean** se aduc trei **Push Button**, patru **Round LED**-uri și un **Vertical Toggle Switch**, cărora li se atribuie etichetele conform fig.5.5;

- se aranjează obiectele din panoul frontal conform fig.5.5, după care se trece în diagrama bloc și se face aceeași operație de aranjare (conform fig.5.4);

- se adaugă în diagramă obiectele lipsă, după cum urmează:

- din paleta **Functions** >> **Analyze** >> **Waveform Generation** se aduce un **Sine Waveform.vi**;
- din paleta **Functions** >> **Comparison** un **Greater Or Equal?**;
- din paleta **Functions** >> **Array** un **Index Array**;
- din paleta **Functions** >> **Boolean** două elemente logice **And**, un **Not** și un **Or**;





În tabelul 5.1, stare 0 la o comandă de pe panoul frontal înseamnă buton neapăsător, iar stare 1 – buton apăsat, în timp ce la dispozitivul extern **SN 7545** s-a simbolizat cu litera S starea inactivă (segment stins) și cu A starea activă (segment aprins).

5.2. Se deschide în **LabVIEW** aplicația **generare compusa si comenzi afisor\_[nume/prenume student]\_[grupa].vi**, efectuându-se aceleași operații preliminare de la punctul 5.1 (se introduce valoarea **1** la controlul **digital channel** și se pornește sursa de alimentare **SA**).

Se lansează în execuție aplicația cu săgeata **Run** de pe bara de comenzi, după care se alege modul **Monoimpuls** pentru controlul de selecție; se inițializează numărătorul binar în “0”, trecând butoanele de comandă **Reset** și **J-K** în “0”, după care se trece butonul **Reset** în “1” și apoi butonul **J-K** în “1”. În acest moment toate cele 4 LED-uri aparținând ieșirilor din **NB** (**Q<sub>A</sub>**, **Q<sub>B</sub>**, **Q<sub>C</sub>**, **Q<sub>D</sub>**) sunt stinse, semnificând starea inițială “0” a acestuia, de asemenea fiind stinse segmentele indicatorului de pe panou, ca și cele ale afișorului **SN 7545** de pe placa **PLC**; prin acționarea butonului **Monoimpuls** de pe panoul frontal, se constată – după fiecare impuls – incrementarea cu o unitate binară atât în **NB** cât și în afișajele de pe panou și dispozitivul **SN 7545**. Pentru fiecare impuls aplicat se trec rezultatele în tabelul 5.2.

Se trece comutatorul de selecție pe poziția **Generator**, se alege o valoare a controlului **Amplit** în jur de **8** și a controlului **Frecv** de aproximativ **0,0020**, constatându-se modificarea – cu o cadență de aproximativ o secundă – a stărilor din **NB**, afișaj local și afișor **SN 7545**. De asemenea, pe ecranul **Waveform Chart**-ului se vor observa mici “întreruperi” ale semnalului sinusoidal, care se datorează momentelor de comutare a stării numărătorului binar **NB**. La terminarea experimentului se oprește aplicația din comanda **STOP** aflată pe panoul frontal.

Tabelul 5.2

Nr.crt. impuls	Stările ieșirilor din numărătorul binar <b>NB</b>				Stările ieșirilor afișaj local/ dispozitiv <b>SN 7545</b>								Cifra afișată
	<b>Q<sub>A</sub></b>	<b>Q<sub>B</sub></b>	<b>Q<sub>C</sub></b>	<b>Q<sub>D</sub></b>	<b>a</b>	<b>b</b>	<b>c</b>	<b>d</b>	<b>e</b>	<b>f</b>	<b>g</b>	<b>pz</b>	
0	0	0	0	0	A	A	A	A	A	A	S	S	<b>0</b>
1	1	0	0	0	A	A	S	S	S	S	S	A	<b>1.</b>
2													
3													
4													
...													

**NOTĂ.** Simbolurile A și S au aceeași semnificație ca la tabelul 5.1.

### 5.6. Concluzii, observații, chestionar

În referatul întocmit la lucrare se va pune accentul pe posibilitățile de generare a semnalelor digitale folosind placa de tip plug-in **KPCI-3102**.

Se va răspunde la următoarele întrebări:

6.1. La aplicația a doua dezvoltată în cadrul lucrării s-a considerat doar situația utilizării unui numărător binar **NB**. Este posibilă transformarea acestuia într-un numărător binar-zecimal? În cazul unui răspuns afirmativ, construiți un **subVI** având acest mod de operare.

6.2. Dacă în aplicația a doua se introduce un control **milliseconds to wait** în panou, respectiv un **Wait Until Next ms Multiple** în diagramă, așa cum s-a procedat în prima aplicație, credeți că va influența cu ceva funcționarea acesteia? Efectuați această operație, rulați noua aplicație și comentați efectul observat pe panoul frontal.

6.3. În afișajele locale ale traductoarelor numerice indicația este împospătată cu o rată de aproximativ 2 secunde. Credeți că se poate realiza un instrument virtual care să înscrie o cifră dorită (între **0** și **9**) atunci când în aplicația a doua se folosește o frecvență mare pe poziția **Generator**? Ce elemente suplimentare sunt necesare?