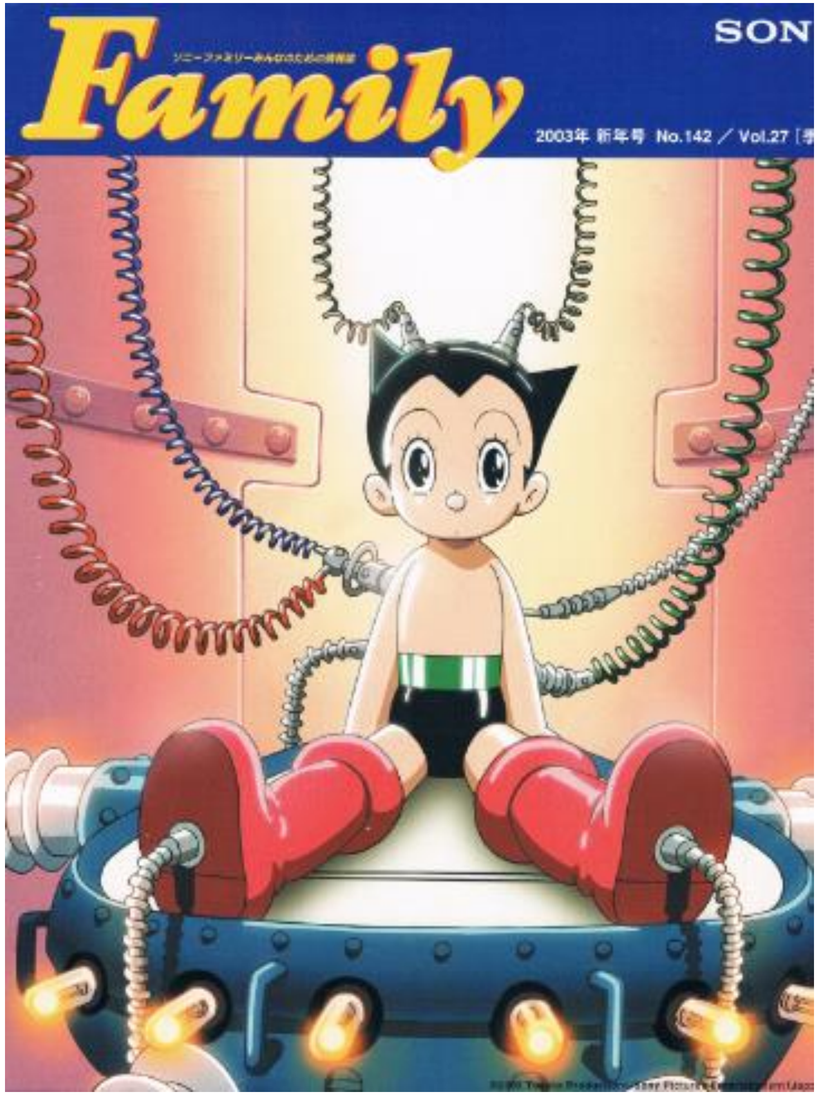
9 Artificial Intelligent Image Sensor

Under Construction

9 Artificial Intelligent Image Sensor

Hagiwara joined Sony in 1975 to work for his dream of building an artificial intelligent (AI) robot system with the smart and real time vision system.

The front page of the Sony Family Journal of January 2003 issue No.142 / Vol 17 shows the Atom Boy.

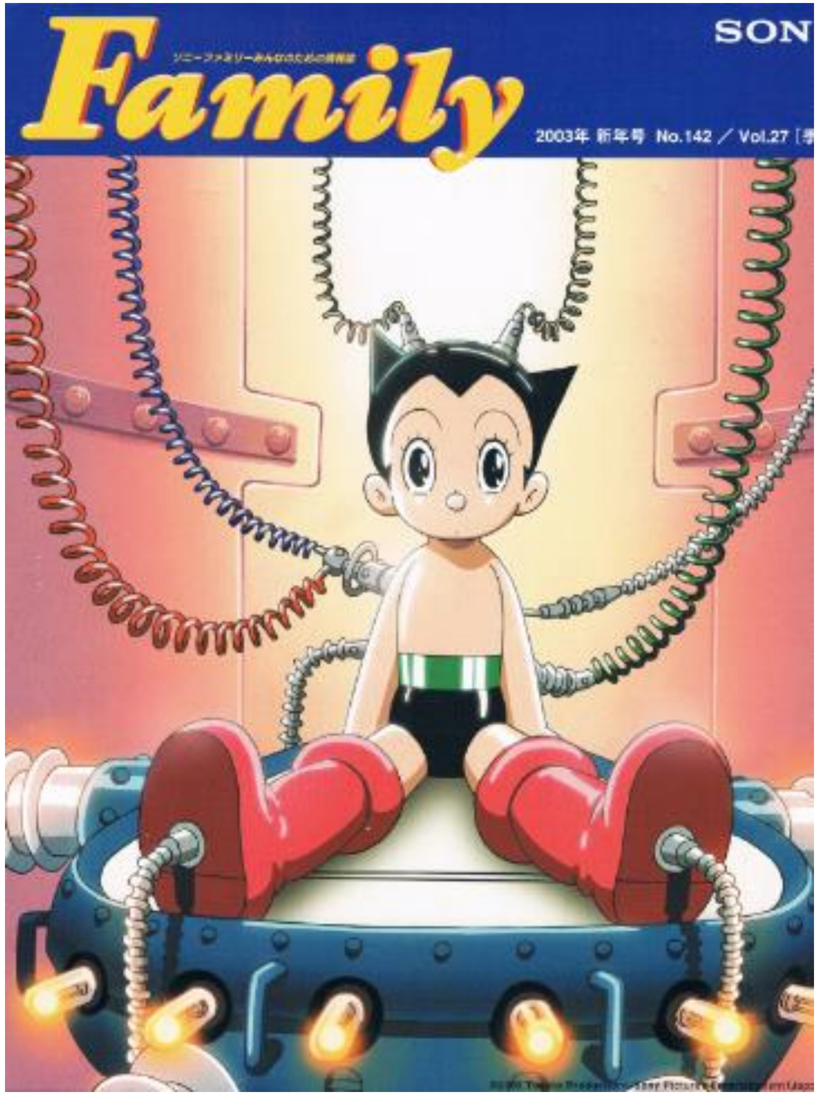

SON
Family
ソニーファミリーみんなのための情報誌
2003年 新年号 No.142 ／ Vol.27 [

9 Artificial Intelligent Image Sensor

Hagiwara worked on CCD image sensors from 1975 till 1980. Then Hagiwara started building the digital CCD camera system.

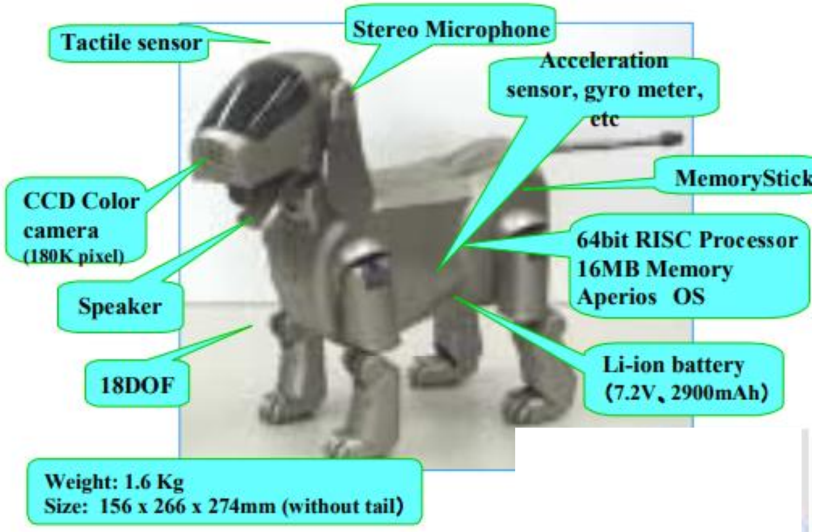
Hagiwara team in Sony in 1989 built the 25 nano sec access time 8 M bit Cache SRAM for fast image data acquisition from CCD image sensor signal output and reported the results at ISSCC1989

Then Hagiwara started working on the micro controller chips, SPC400 and SPC800 series for the Sony consumer products.



9 Artificial Intelligent Image Sensor

In 2001 Hagiwara was invited in the International Conference ESSCIRC2001 in Vilach, Austria to talk about his dream on the entertainment consumer products including the AIBO robot system which was made possible with many important semiconductor components including the intelligent image sensor chip.


Tactile sensor
Stereo Microphone
Acceleration sensor, gyro meter, etc
MemoryStick
CCD Color camera (180K pixel)
64bit RISC Processor 16MB Memory Aperios OS
Speaker
Li-ion battery (7.2V, 2900mAh)
18DOF
Weight: 1.6 Kg Size: 156 x 266 x 274mm (without tail)

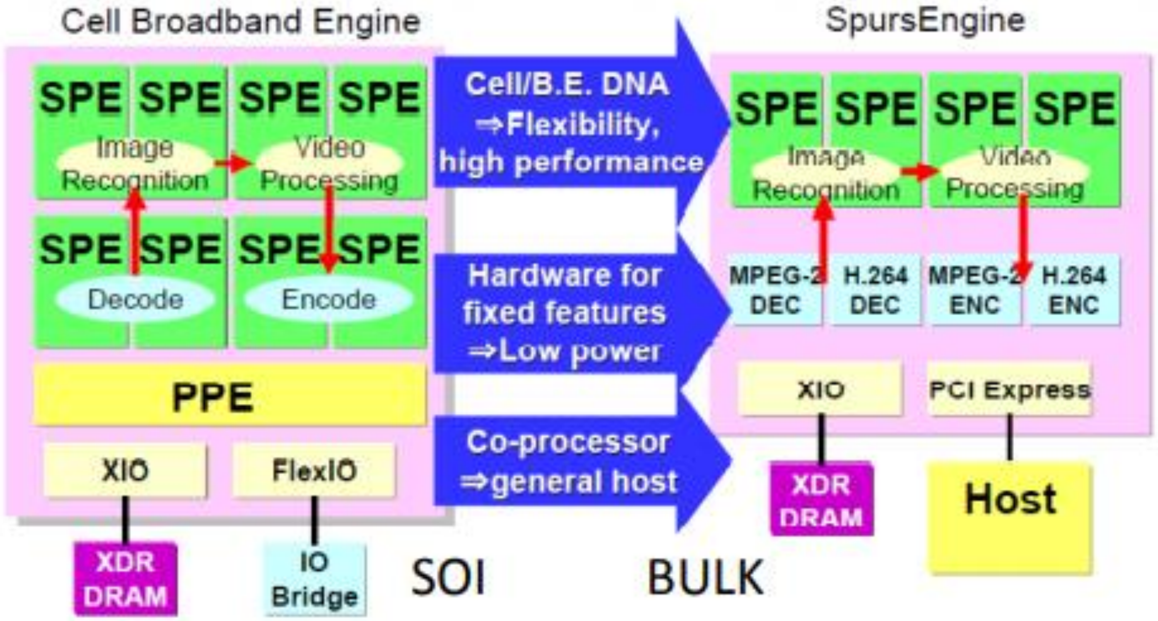
AIBO Model ERS-110

AIBO Model ERS-210

AIBO Model SDR-3

9 Artificial Intelligent Image Sensor

In 2008 Hagiwara was invited in the International Conference ESSCIRC2008 in Edinburgh, Scotland UK , to talk about his dream on the entertainment consumer products including the real time fast dedicated cell processor engines, the Cell / B. E . and Toshiba Spurs, for the PS3 game machine.

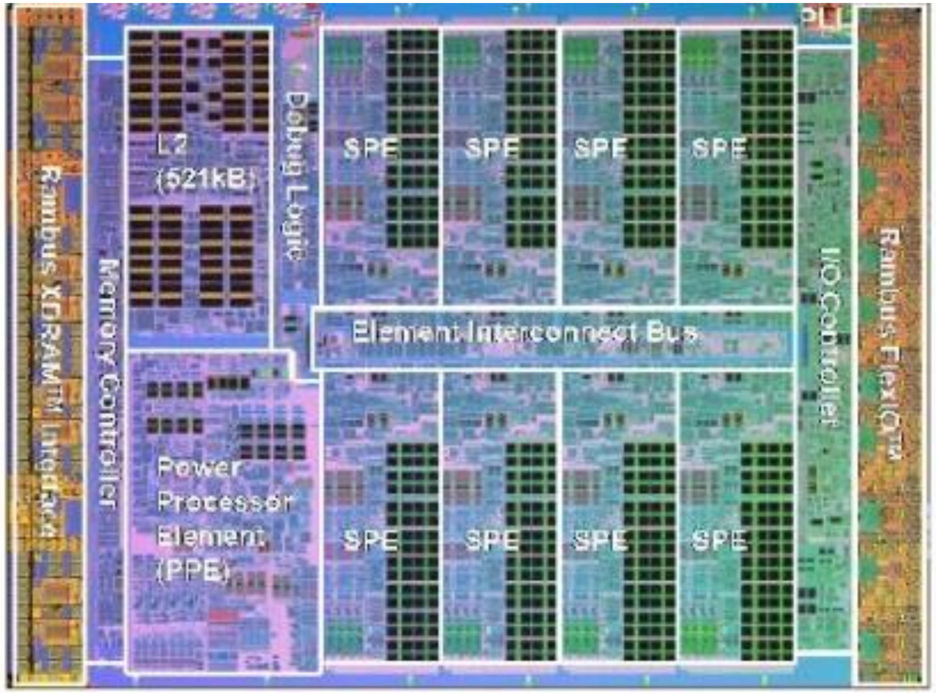


Sony/IBM/Toshiba Cell/B.E. and Toshiba Spurs Engine

Mitsuo Saito at ICD-ARC Panel, May 13, 2008

9 Artificial Intelligent Image Sensor

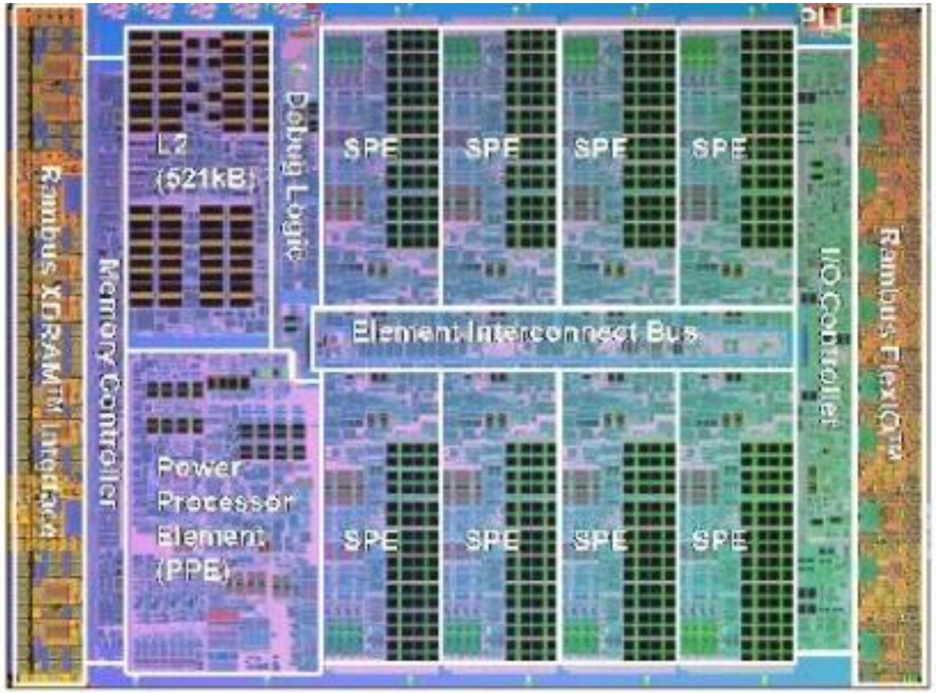
Software approach of Real Time Pattern Recognition is also one of the AISP efforts. However, to utilize real time software approach effectively, we need to design first a powerful hardware real time engine, which is much more powerful than Intel processors and Cell processors.



45 nm Cell/B.E. Die Photo by Osamu Takahashi, ISSCC2006

9 Artificial Intelligent Image Sensor

A powerful hardware real time engine needs first a very fast memory unit and a very powerful ALU unit in parallel architecture.



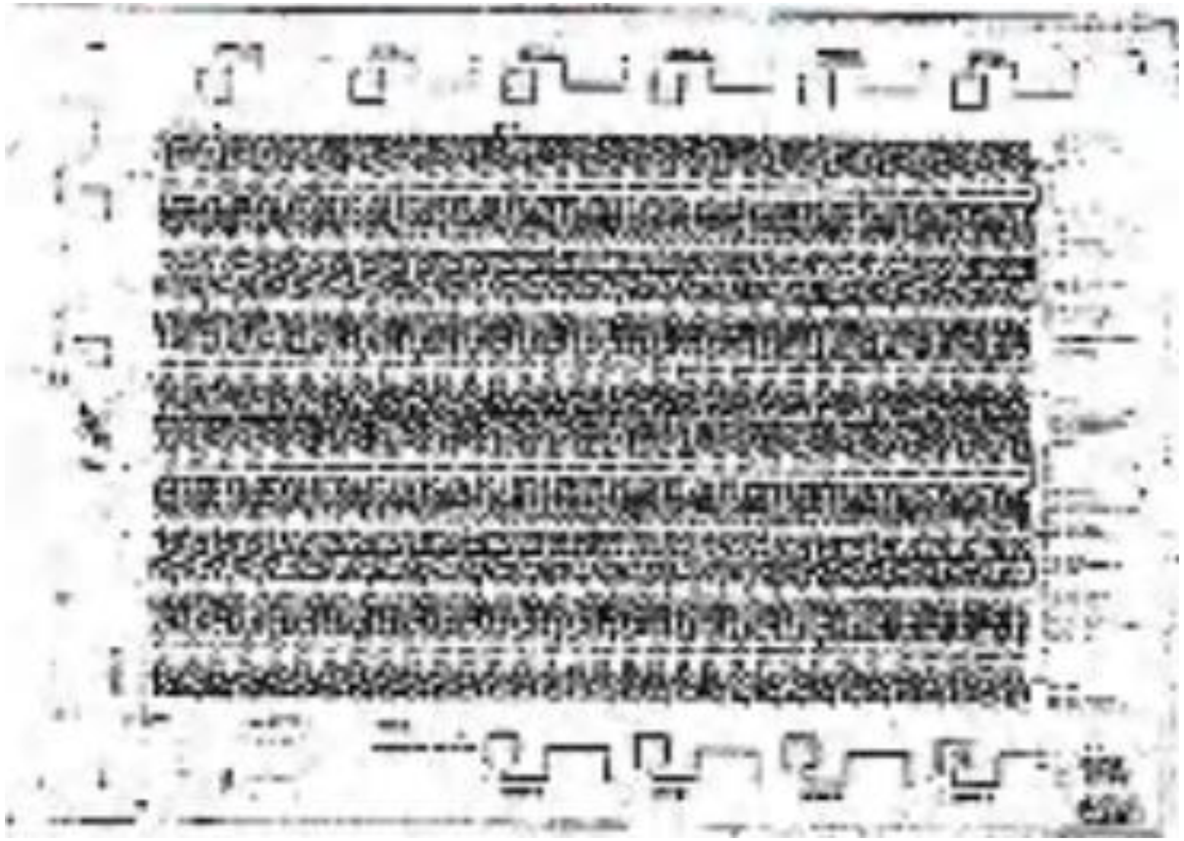
45 nm Cell/B.E. Die Photo by Osamu Takahashi, ISSCC2006

9 Artificial Intelligent Image Sensor

Artificial Intelligent Image Sensor needs a powerful hardware real time engine.

Prof. C. A. Mead and Yoshiaki Daimon Hagiwara working on the silicon chip design at Caltech in 1972

128-bit Multicomparator chip, designed by Hagiwara in 1972-1973 and fabricated by Intel PMOS process.

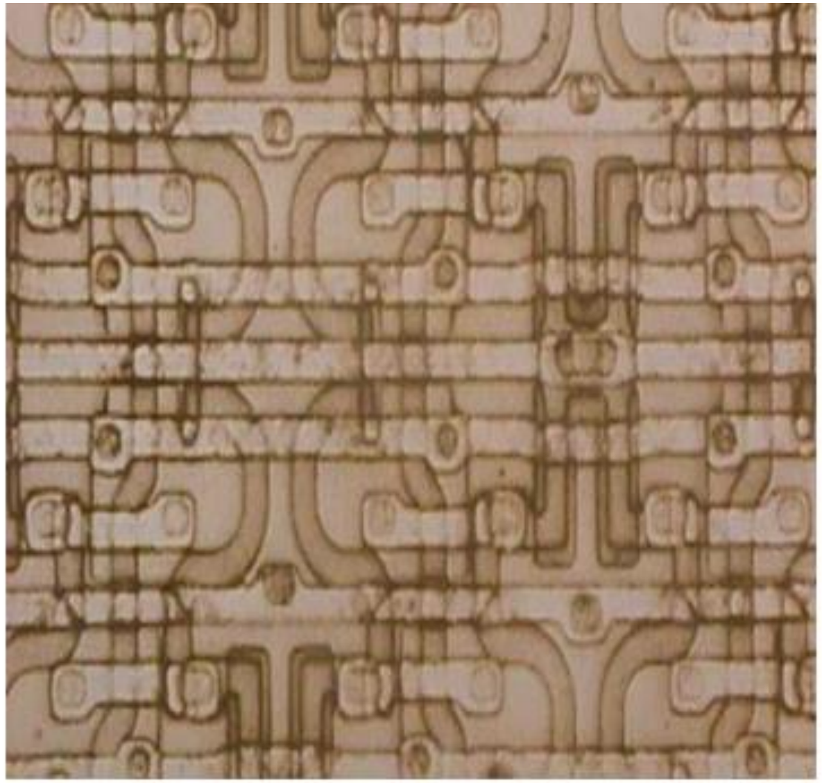


128 bit Multi Comparator Chip

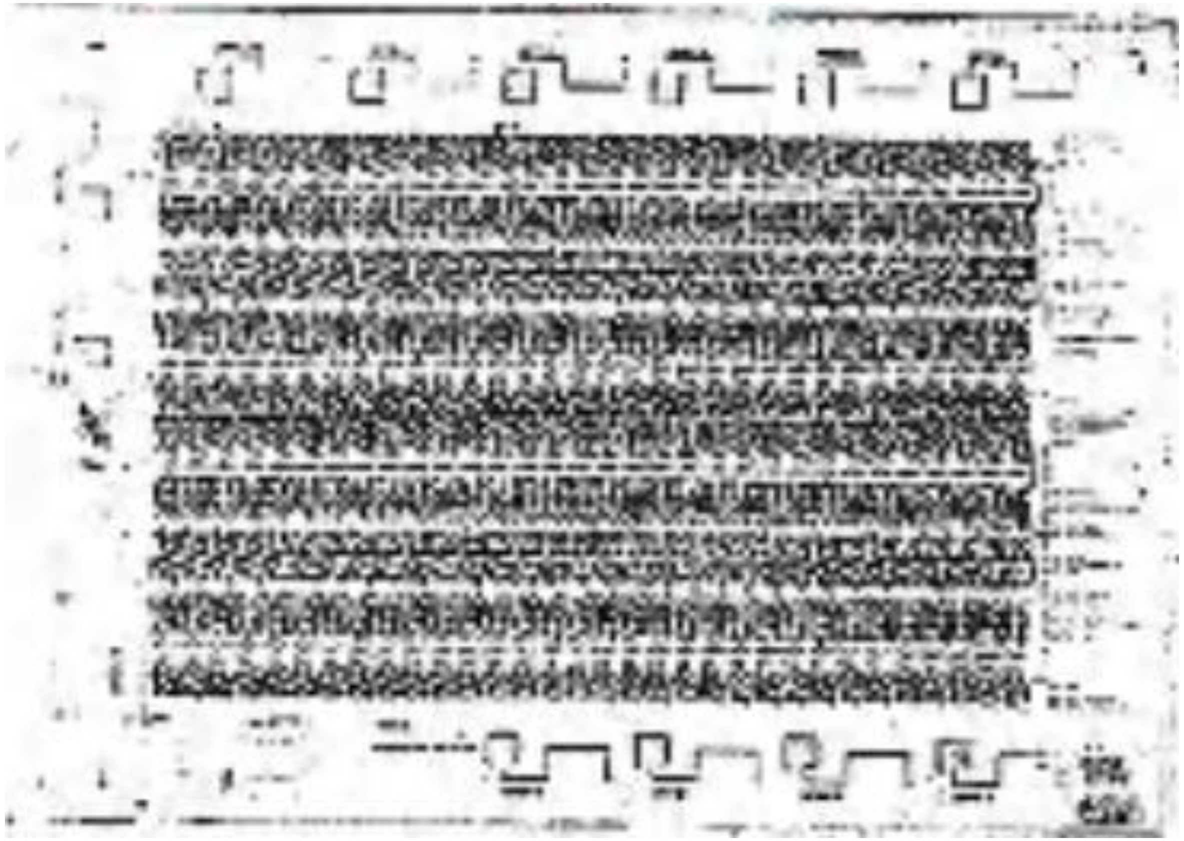
128 bit Multi Comparator Chip designed by Caltech Students and fabricated by Intel Ppocess.
IEEE Journal of Solid State Circuits, VOL.SC11, No.4, October 1976

9 Artificial Intelligent Image Sensor

Artificial Intelligent Image Sensor needs a powerful hardware real time engine.



1971 Intel 1101 256 bit DRAM chip

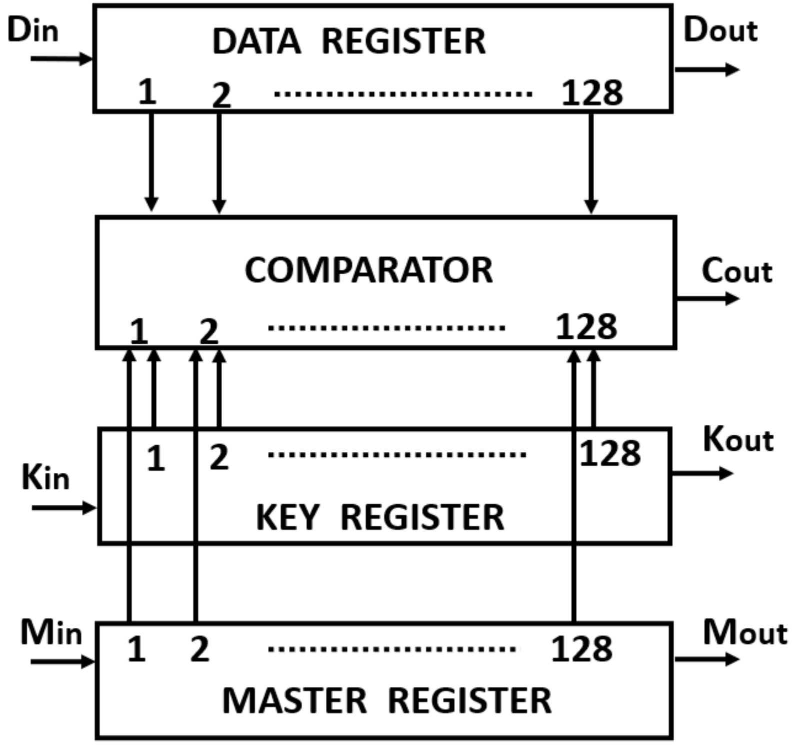


128 bit Multi Comparator Chip

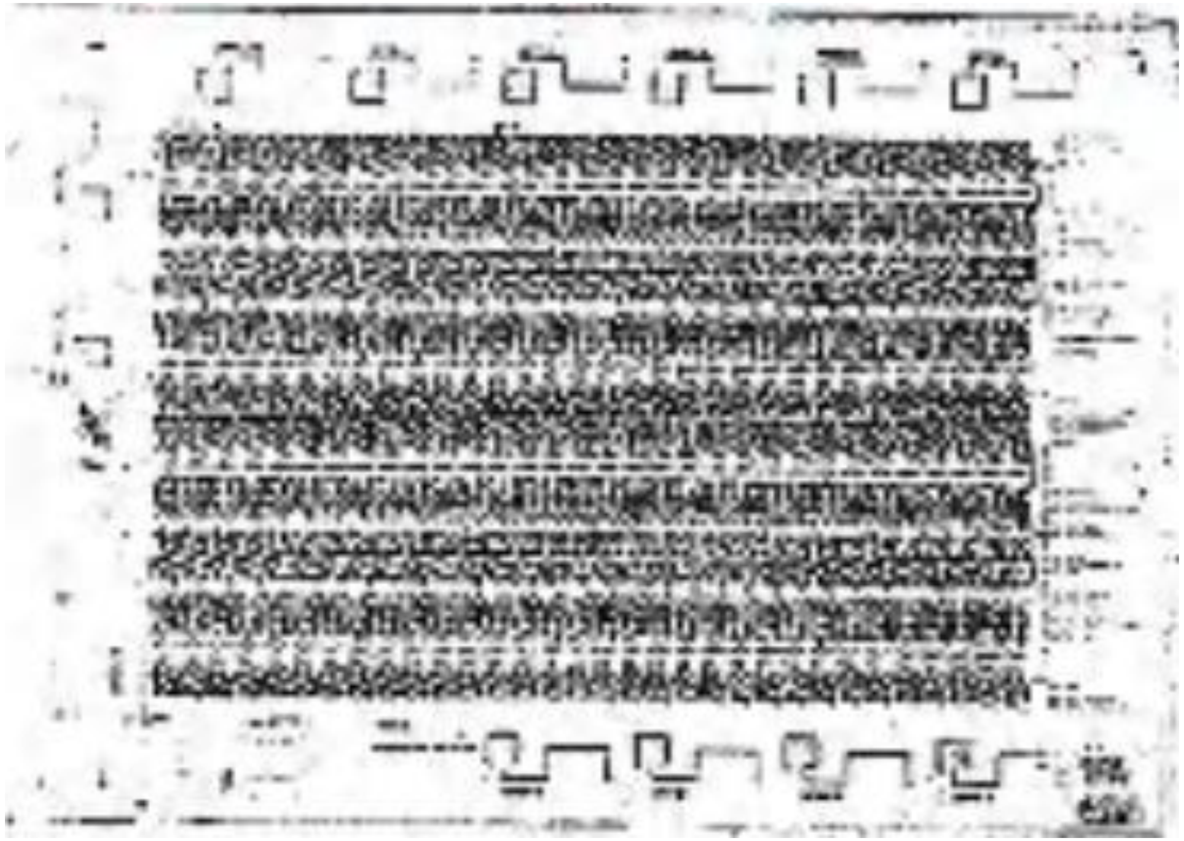
128 bit Multi Comparator Chip designed by Caltech Students and fabricated by Intel DRAM Process.
IEEE Journal of Solid State Circuits, VOL.SC11, No.4, October 1976

9 Artificial Intelligent Image Sensor

Artificial Intelligent Image Sensor needs a powerful hardware real time engine.



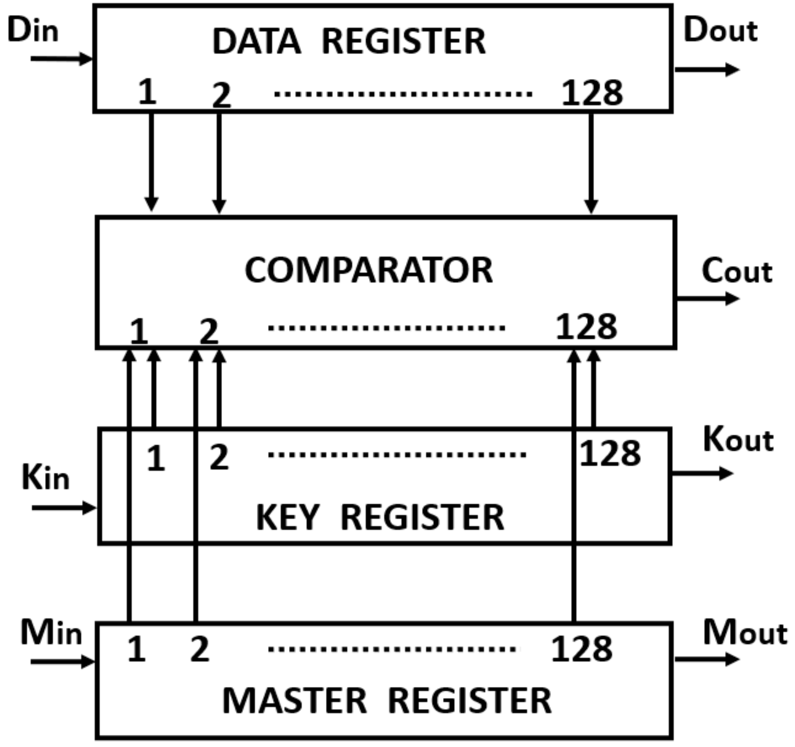
Block Diagram of 128 bit Multi Comparator



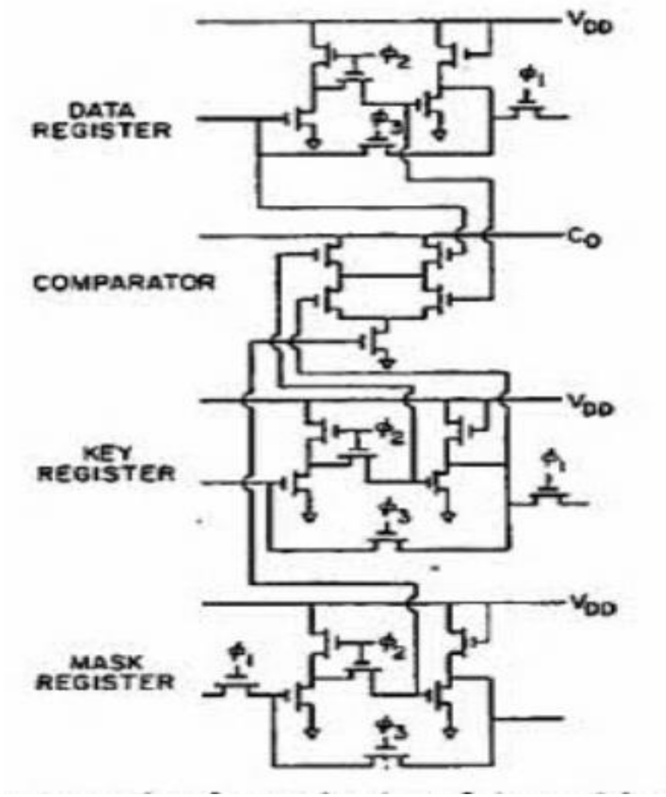
128-Bit Multicomparator Chip designed by Caltech Students and fabricated by Intel.
IEEE Journal of Solid State Circuits, VOL.SC11, No.4, October 1976

9 Artificial Intelligent Image Sensor

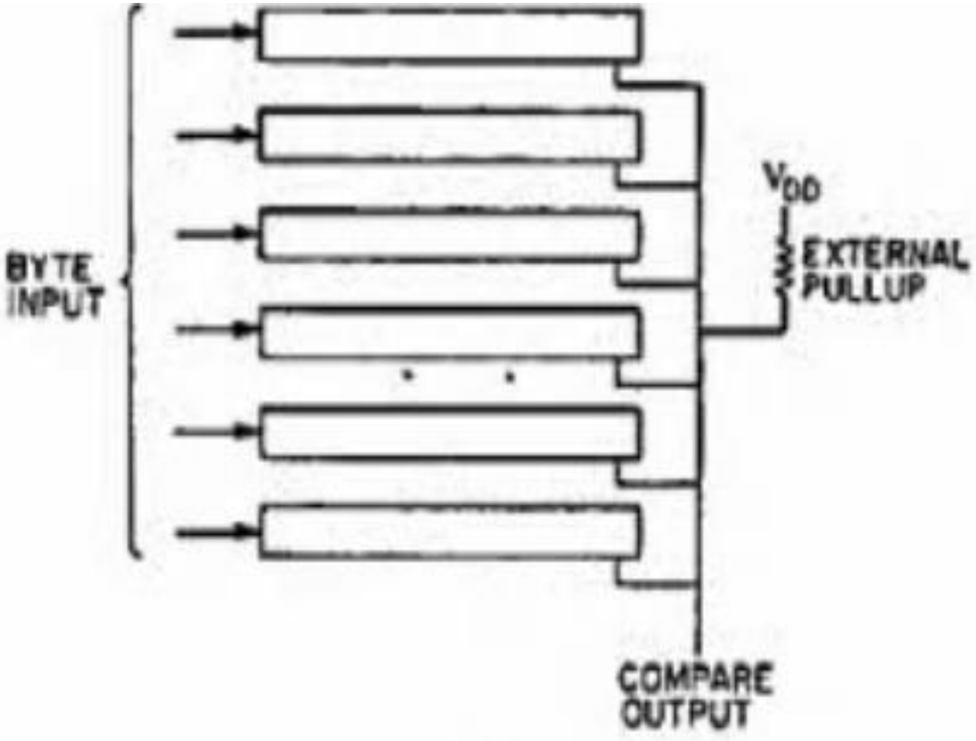
Artificial Intelligent Image Sensor needs a powerful hardware real time engine.



Block Diagram of 128 bit Multi Comparator



Full Schematic of One Bit Slice of the Multi Comparator

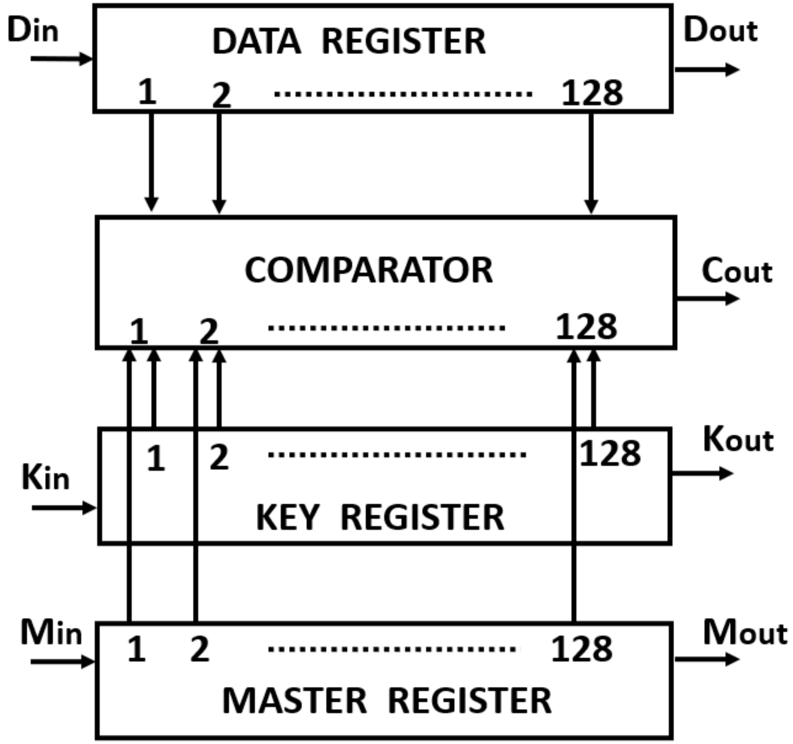


Bit Paralle, Word Serial Organization

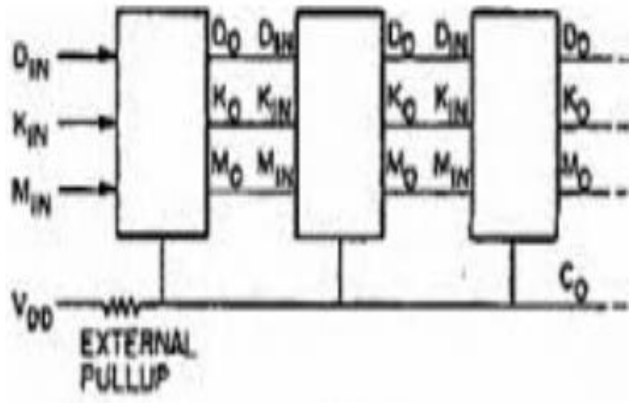
128-Bit Multicomparator Chip designed by Caltech Students and fabricated by Intel.
IEEE Journal of Solid State Circuits, VOL.SC11, No.4, October 1976

9 Artificial Intelligent Image Sensor

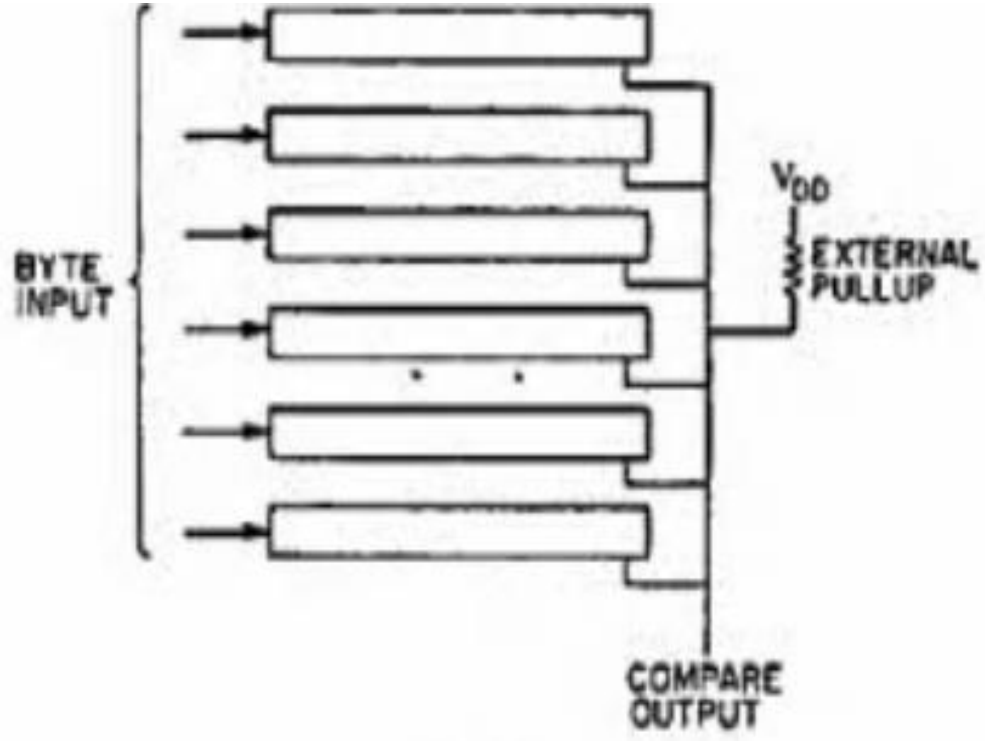
Artificial Intelligent Image Sensor needs a powerful hardware real time engine.



Block Diagram of 128 bit Multi Comparator



Cascaded Organization

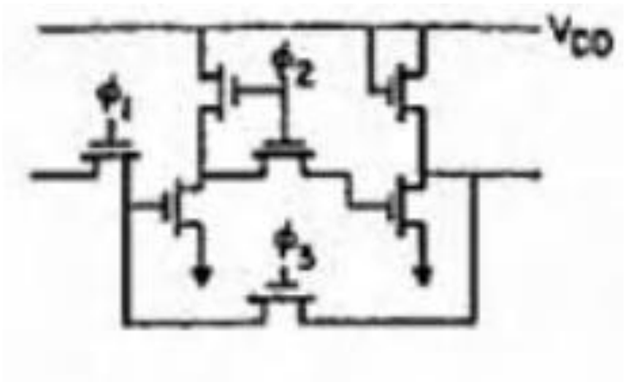


Bit Paralle, Word Serial Organization

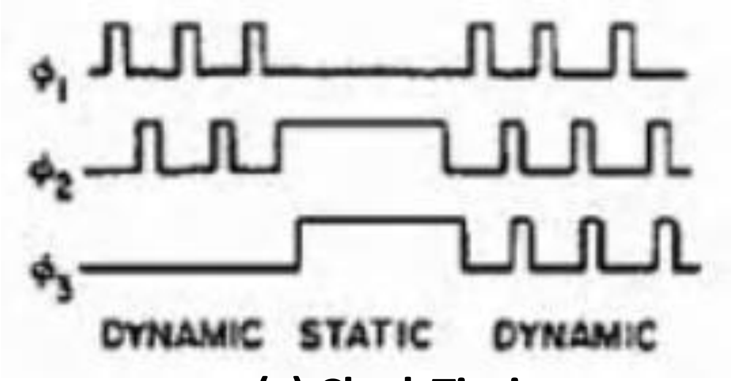
128-Bit Multicomparator Chip designed by Caltech Students and fabricated by Intel.
IEEE Journal of Solid State Circuits, VOL.SC11, No.4, October 1976

9 Artificial Intelligent Image Sensor

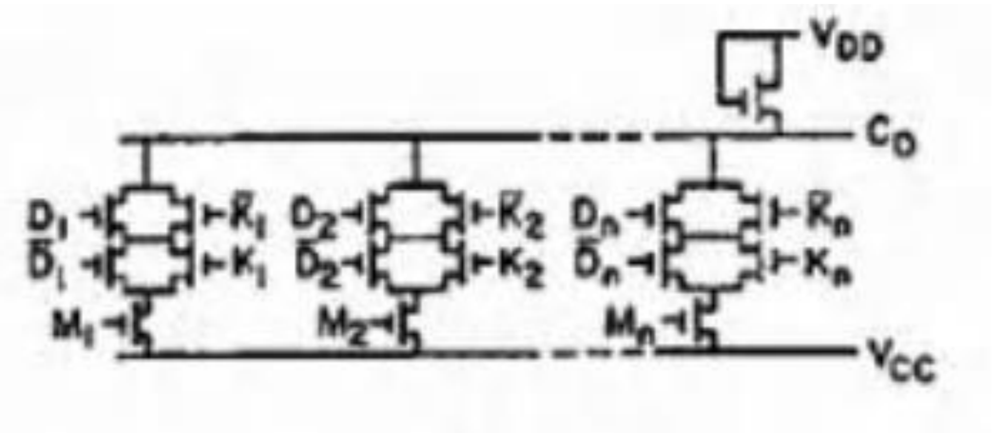
Artificial Intelligent Image Sensor needs a powerful hardware real time engine.



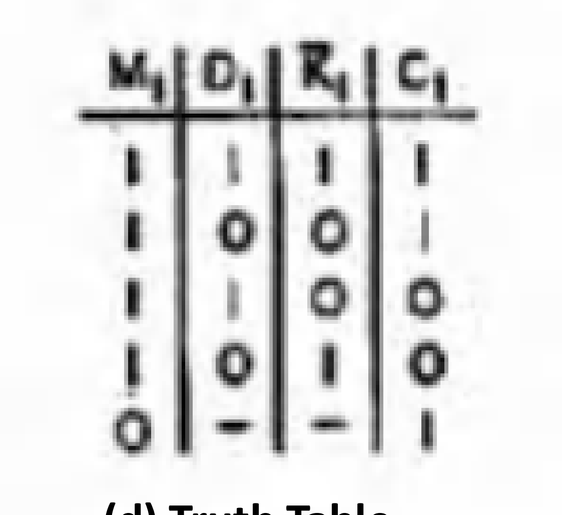
(a) Basic Shift Register



(c) Clock Timing

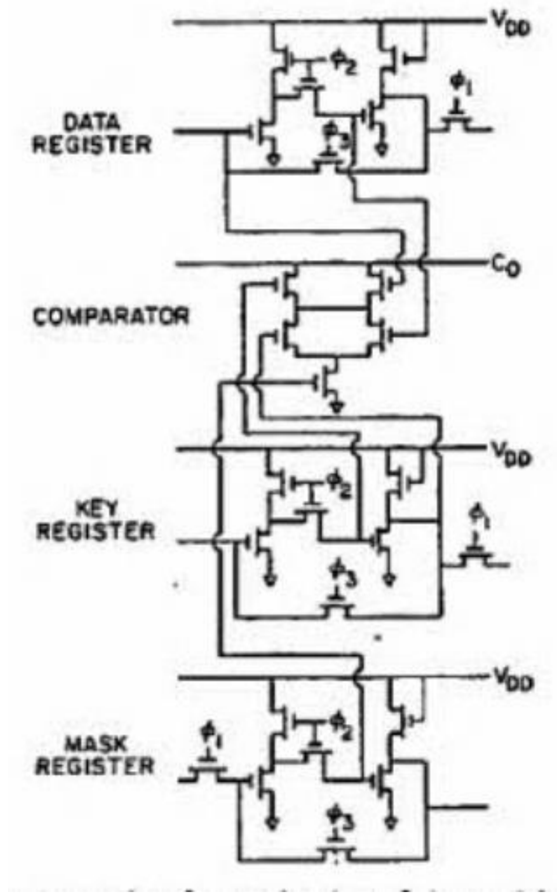


(b) Gated Exclusive Nor Gate



M_1	D_1	R_1	C_1
1	1	1	1
1	0	0	1
1	1	0	0
1	0	1	0
0	–	–	1

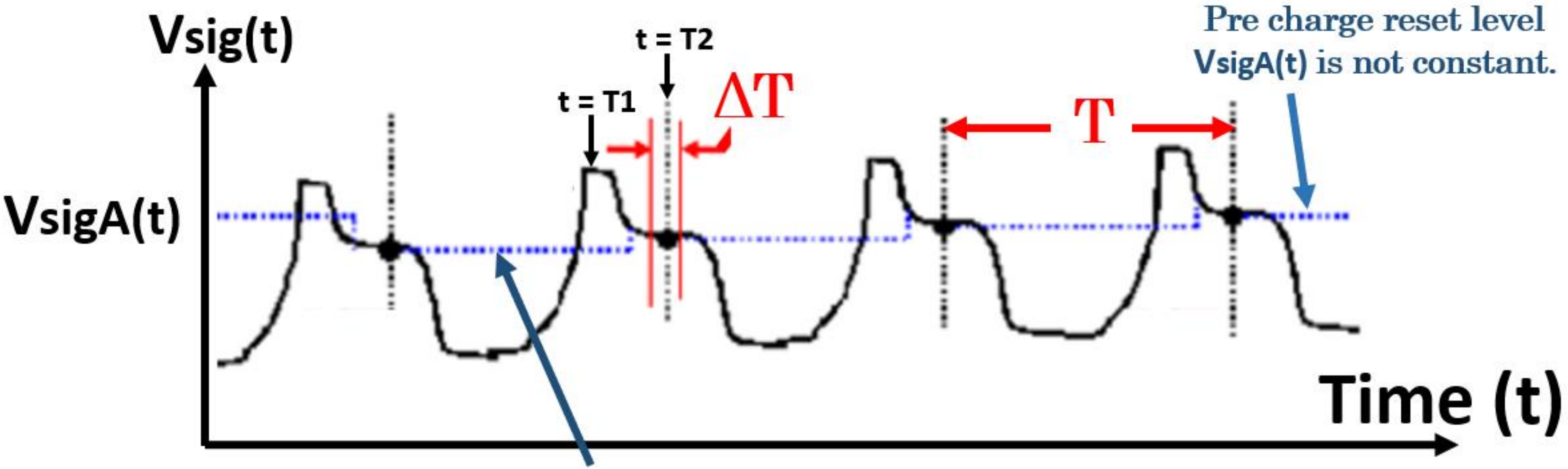
(d) Truth Table



Full Schematic of One Bit Slice of the Multi Comparator

9 Artificial Intelligent Image Sensor

The output signal of an image sensor is buried in many kinds of noises and appears only in a short time slot ΔT during the total cycle time T.

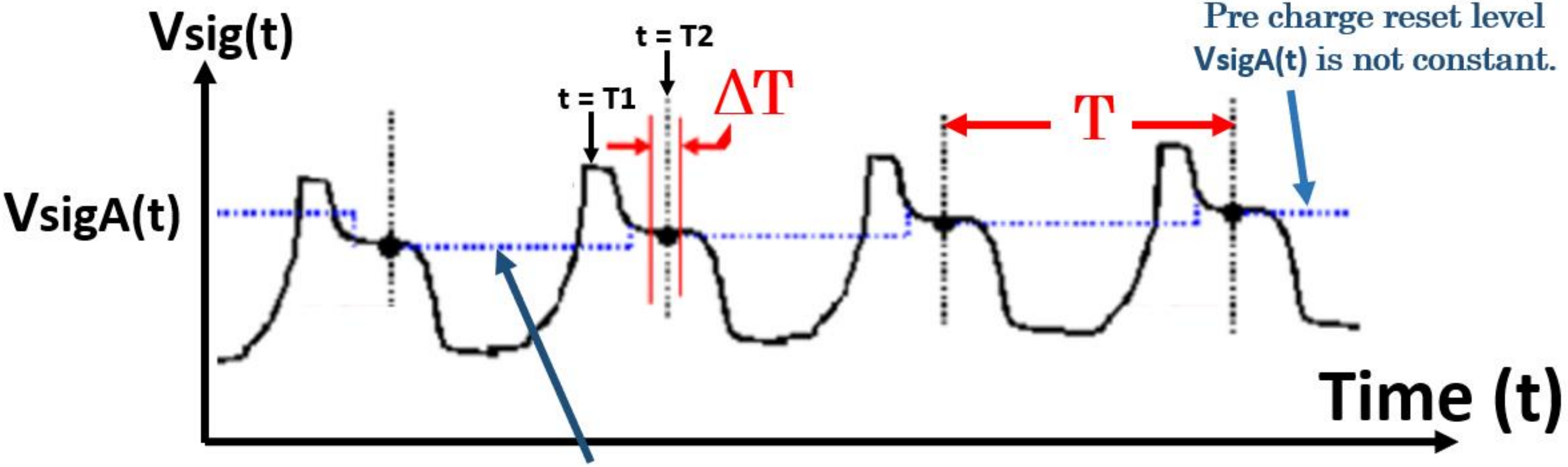


Pre charge reset level VsigA(t) is not constant.

The output signal VsigA(t) of the Single Sample Hold Circuit still includes a lot of noise component.

9 Artificial Intelligent Image Sensor

Our task is extracting the important information needed for us from the image sensor output signal which has many kinds of noise.

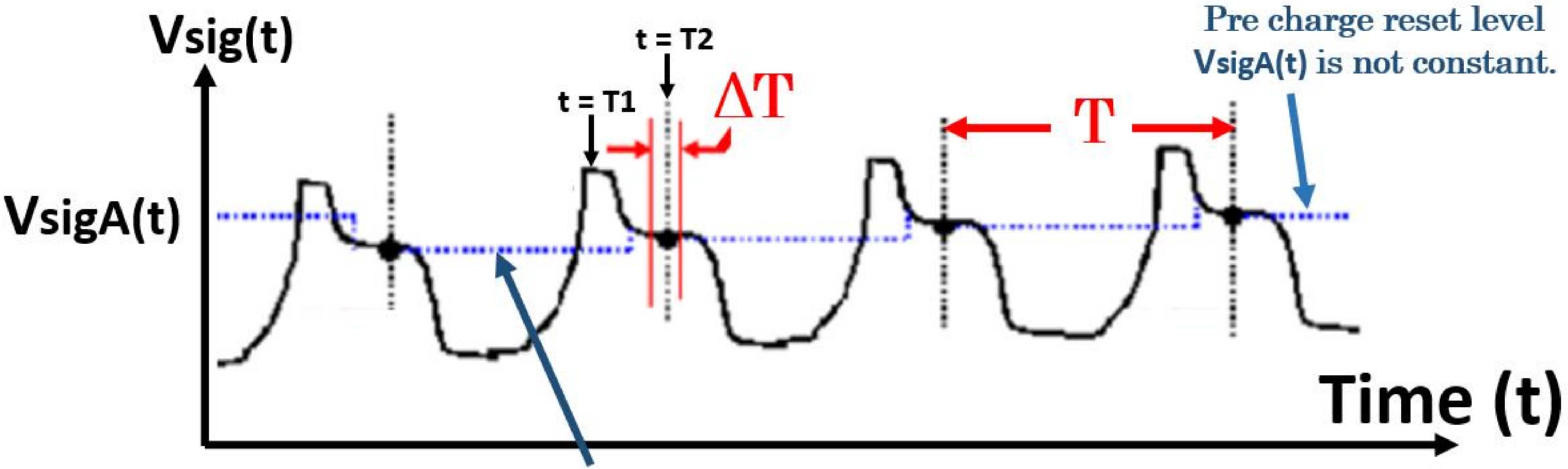


Pre charge reset level VsigA(t) is not constant.

The output signal VsigA(t) of the Single Sample Hold Circuit still includes a lot of noise component.

9 Artificial Intelligent Image Sensor

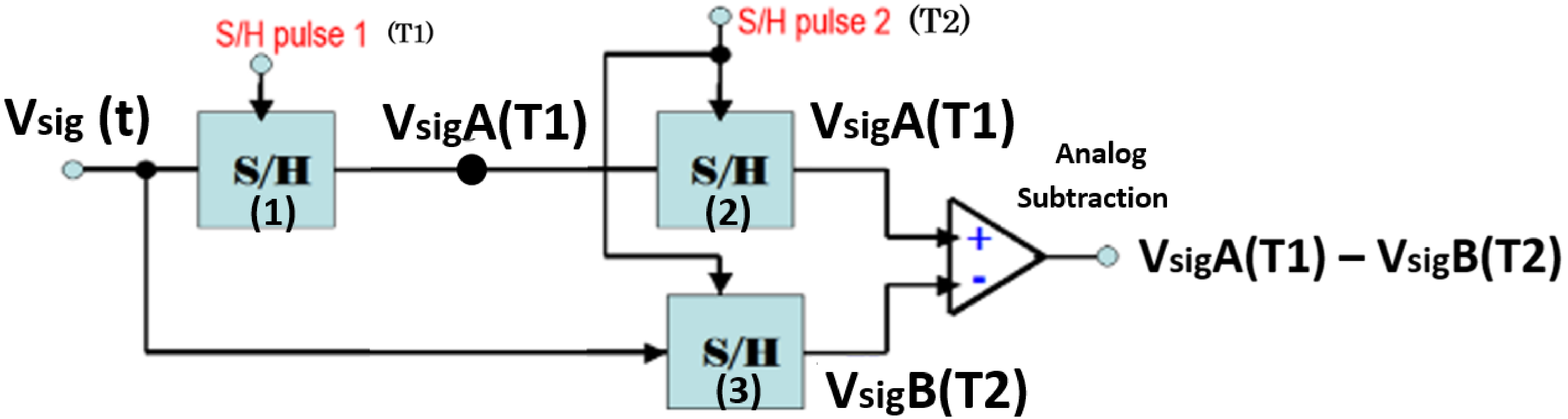
The effort of extracting the important information needed for us is by definition the artificial intelligent signal processing (AISP).



The output signal VsigA(t) of the Single Sample Hold Circuit still includes a lot of noise component.

9 Artificial Intelligent Image Sensor

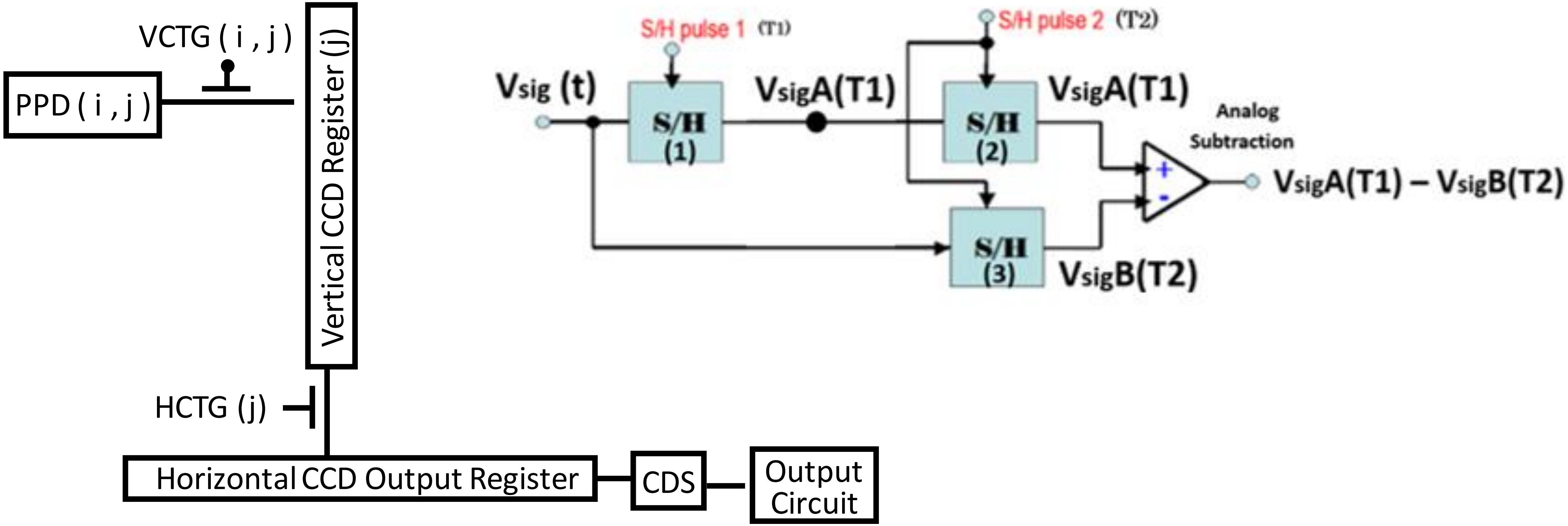
In 1972, M. White proposed Correlated Double Sample (CDS) circuit which has three sample hold circuits and one analog subtraction circuit to delete the undesired noise components of the output signal of CCD image sensors.



M .H. White, D. R. Lanpe, F. C. Blaha and I. A . Mack,
"Characterization of surface Channel CCD Image Arrays at Low Light Level",
IEEE Journal of Solid State Circuits, SC-9, pp.1-13 (1974)

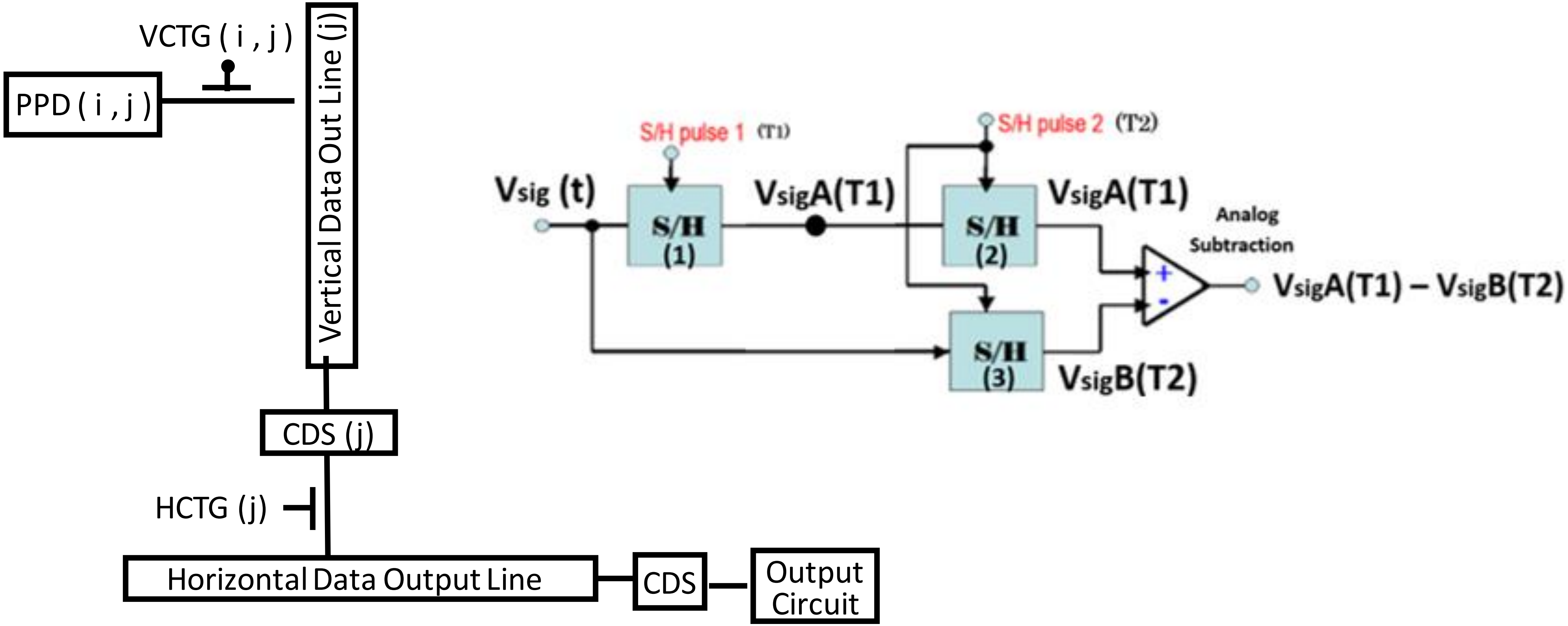
9 Artificial Intelligent Image Sensor

Only one CDS circuit was originally used in classical CCD image sensors.



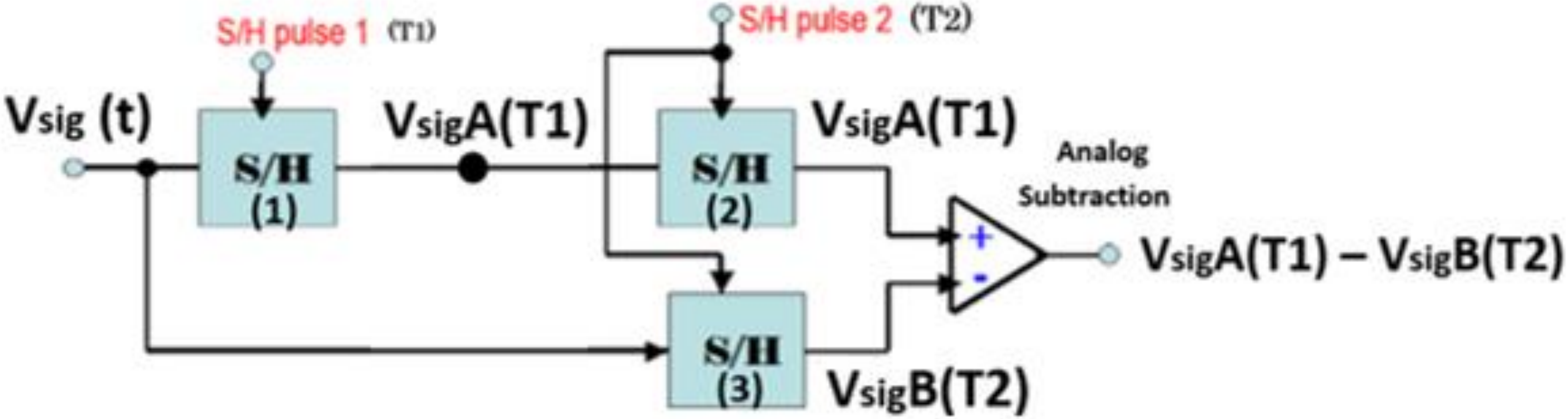
9 Artificial Intelligent Image Sensor

But now one CDS circuits was built in each column line.



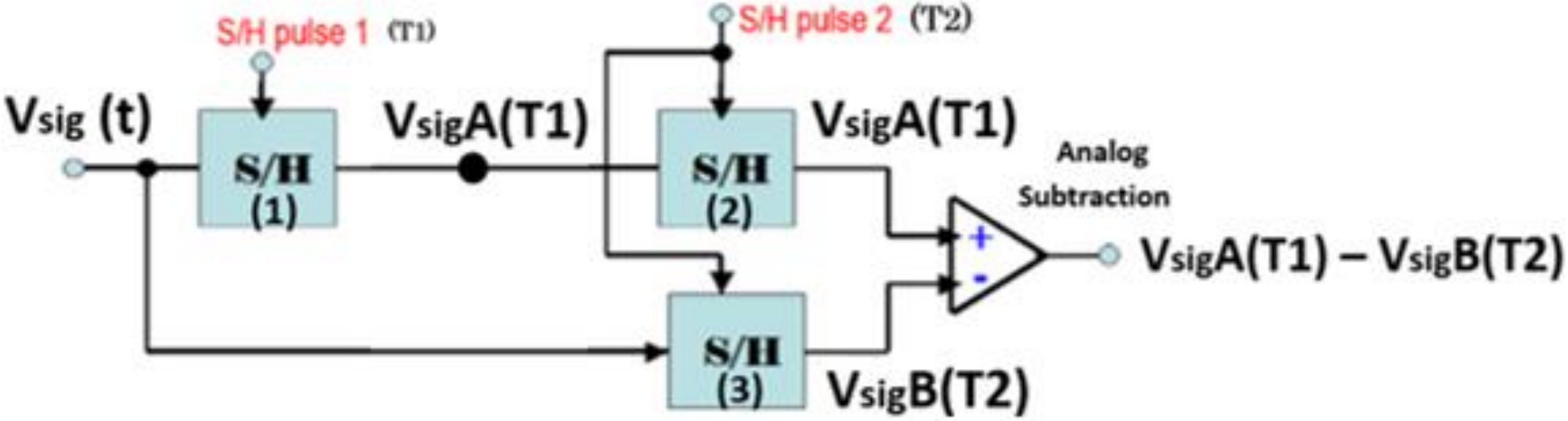
9 Artificial Intelligent Image Sensor

The correlated double sample (CDS) circuit needs a powerful hardware engine to perform analog signal subtraction function which must be implemented by a well balanced and carefully designed differential amplifier circuit built with the CMOS process technology.



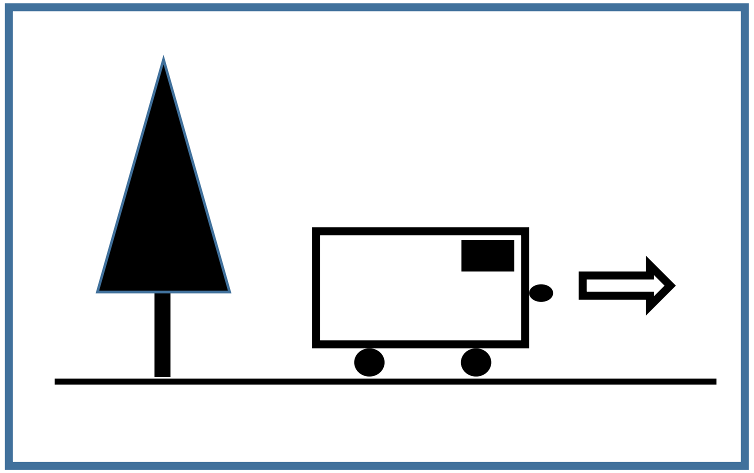
9 Artificial Intelligent Image Sensor

The intelligent circuit design of the correlated double sampling hold (CDS) is also one of the AISP efforts.

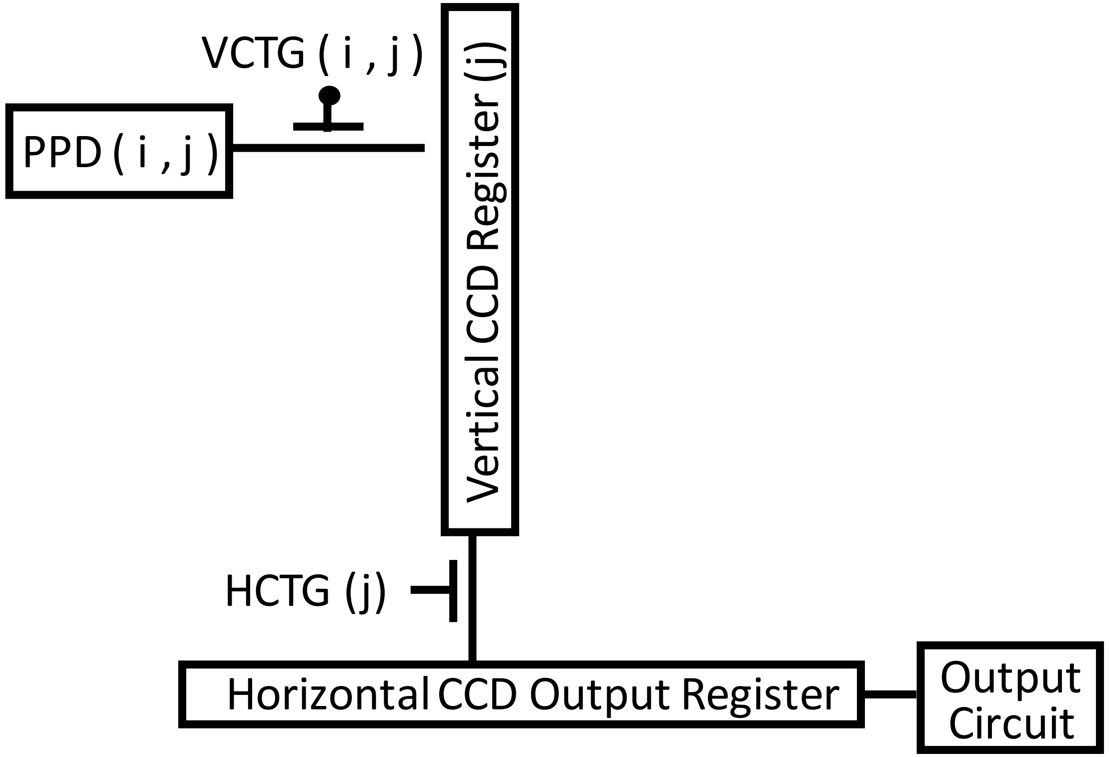


9 Artificial Intelligent Image Sensor

The CCD image sensors with the CCD type CTD was built with the vertical and horizontal CCD type analog shift registers which are made of many CCD/MOS memory capacitors acting also as the in pixel Global Shutter Buffer Memory (GSBM).

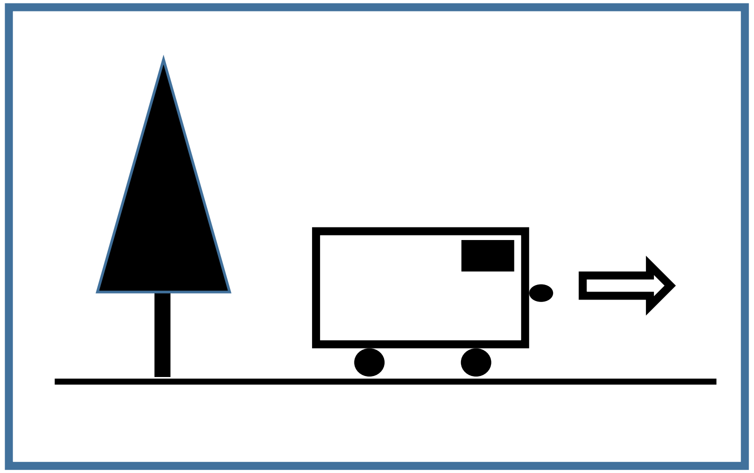


the CCD image sensors with the Built in Global Shutter Function

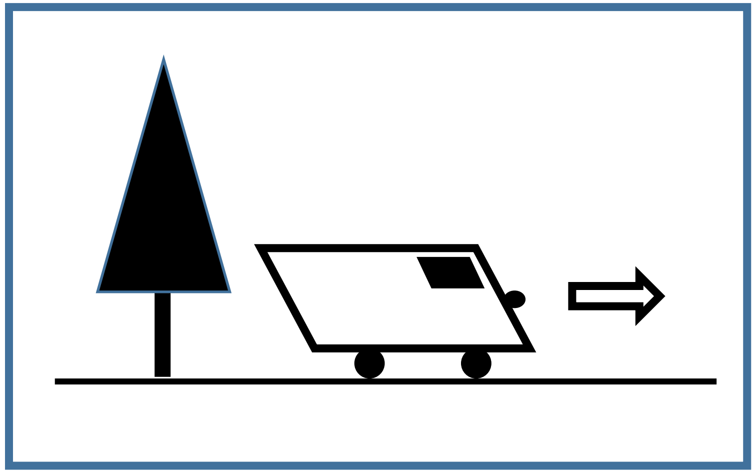


9 Artificial Intelligent Image Sensor

However, the classical CMOS image sensors used the CMOS type charge transfer device (CTD) which had a problem of image distortion in moving objects by the rotary shutter effect.



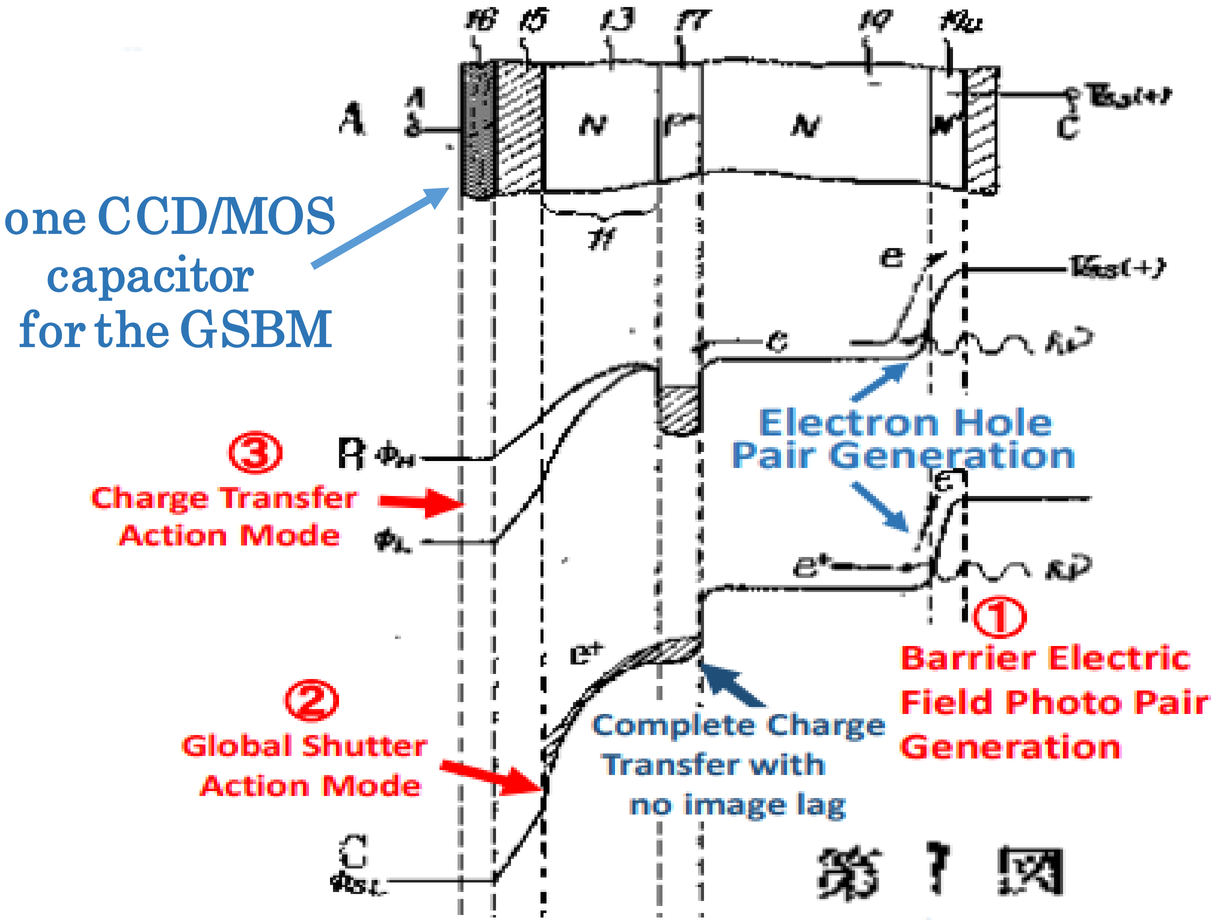
the CCD image sensors with the Built in Global Shutter Function



the classical CMOS image sensors with rotary shutter effect

9 Artificial Intelligent Image Sensor

The in pixel Global Shutter Buffer Memory (GSBM) was needed absolutely in CMOS image sensors. Hinted by the CCD type CTD with the built in CCD/MOS capacitor type GSBM, Hagiwara in 1975 proposed to add one CCD/MOS capacitor type GSBM in each pixel with the three voltage level CTG operation scheme.



one CCD/MOS capacitor for the GSBM

③ Charge Transfer Action Mode

Electron Hole Pair Generation

① Barrier Electric Field Photo Pair Generation

② Global Shutter Action Mode

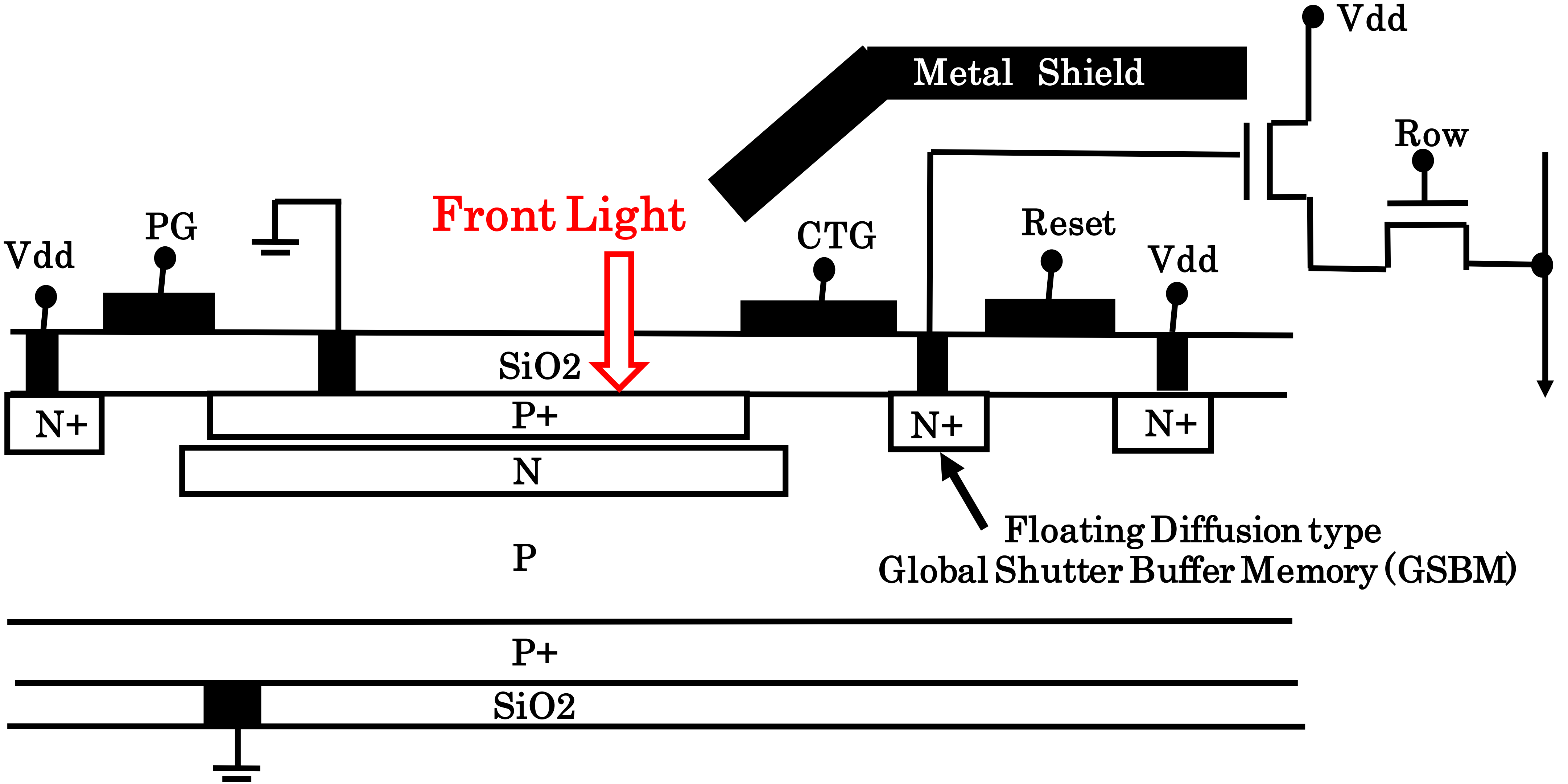
Complete Charge Transfer with no image lag

Fig. 7

Yoshiaki. Hagiwara, Japanese Patent Applications JPA 1975-127647, JPA 1975-127647 and JPA 1975-134985.

9 Artificial Intelligent Image Sensor

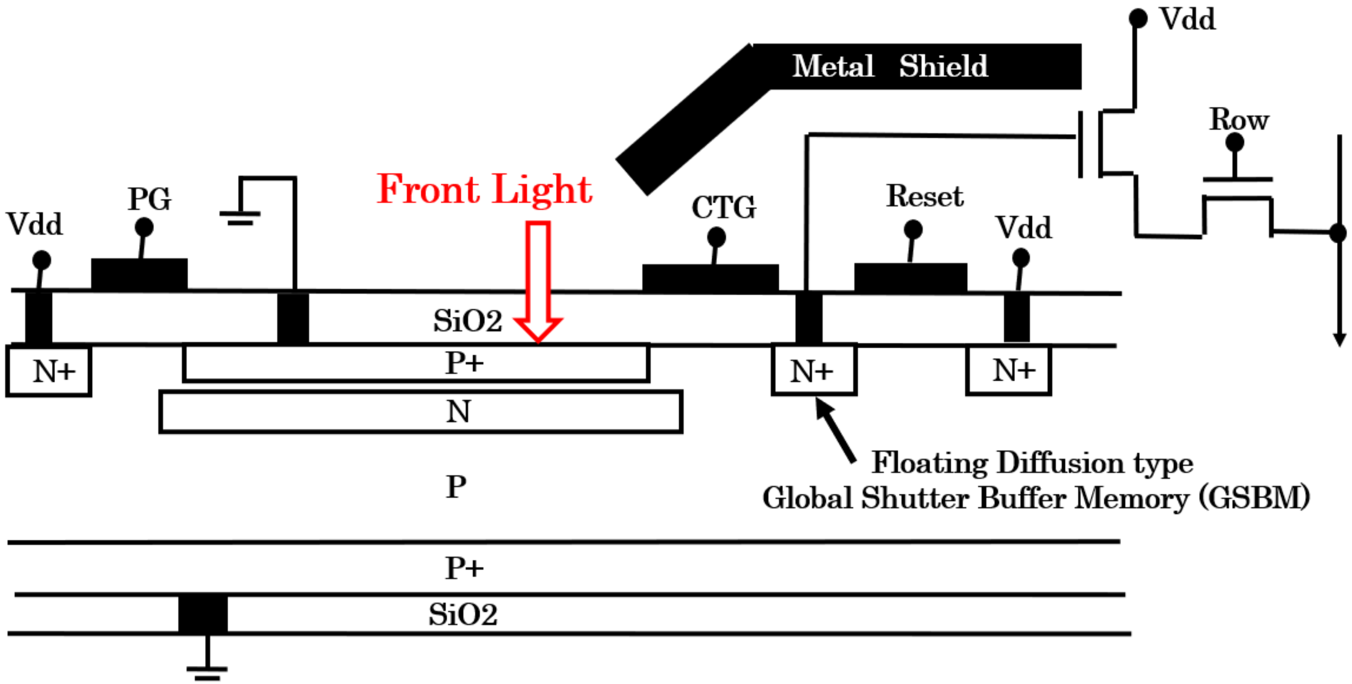
1T1C DRAM has the N+ floating source diffusion as the memory capacitor.



Floating Diffusion type
Global Shutter Buffer Memory (GSBM)

9 Artificial Intelligent Image Sensor

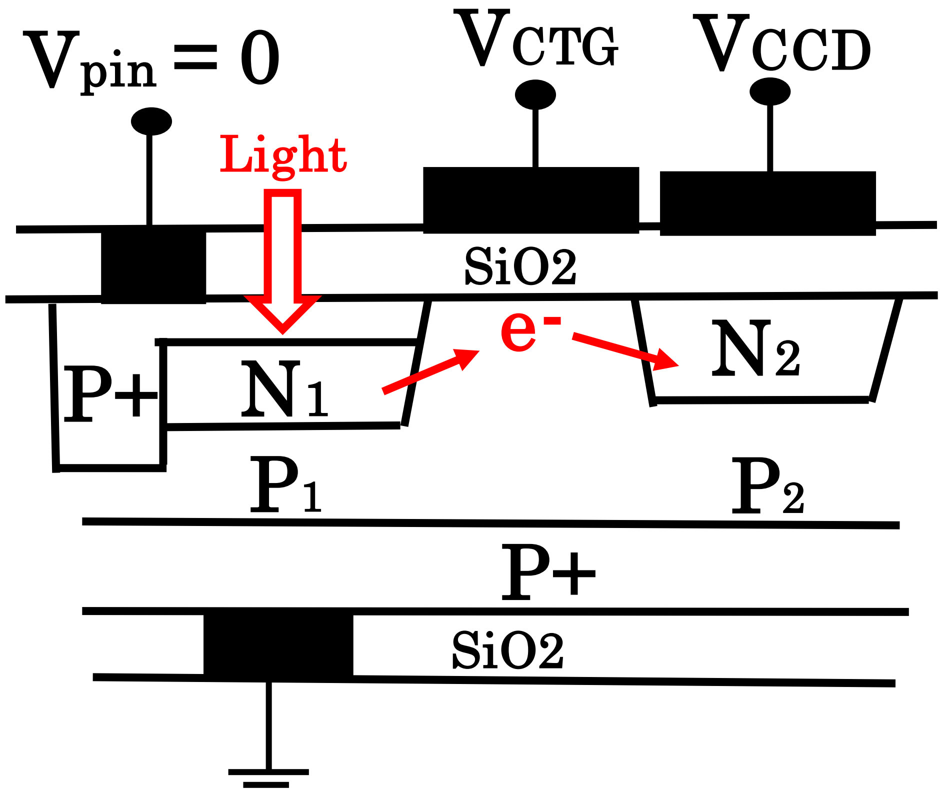
It is a natural choice to use also the floating diffusion for the GSBM for CMOS type CTD image sensors.



Guang Yang, Orly Yadid-Pecht, Chris Wrigley, and Bedabrata Pain,
"A Snap-Shot CMOS Active Pixel Imager for Low-Noise, High-speed Imaging",
Technical Journal of IEDM1998, 98-45, Dec 1998.

9 Artificial Intelligent Image Sensor

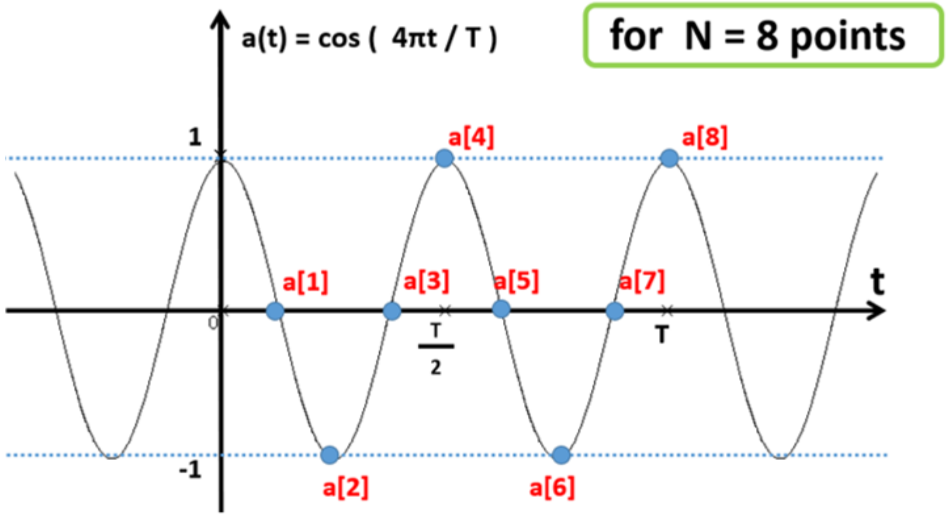
The intelligent device design of the Pinned Photodiode (PPD) with low noise and very excellent light sensitivity is one of the AISP efforts.



9 Artificial Intelligent Image Sensor

Artificial Intelligent Image Sensor needs many kinds of powerful hardware real time engines. Discrete Fourier Transformation (DFT) Engine is one of them which is essential for voice and picture recognition.

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme



Sampling Points are at t[k] = kT/8 for k=1 to 8 ;

a[ ] ={ a[1], a[2], ········ , a[8] }

A[ ] ={A[1], A[2], ... , A[8] }

DFT [m][n] = exp (− 2 m n π i / N)

for m = 1, 2, ····· N and n = 1, 2,····· N.

$i^2 = -1$

A[ ] = DFT[ ][ ]a[ ]

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

Design and Performance Estimation of DFT Processing Circuits

Japanese Patent Application JPA 2014-135497

filed on July 1, 2014 by Yoshiaki Hagiwara.

″The digital circuit engine design to convert the time domain information to the frequency domain information″

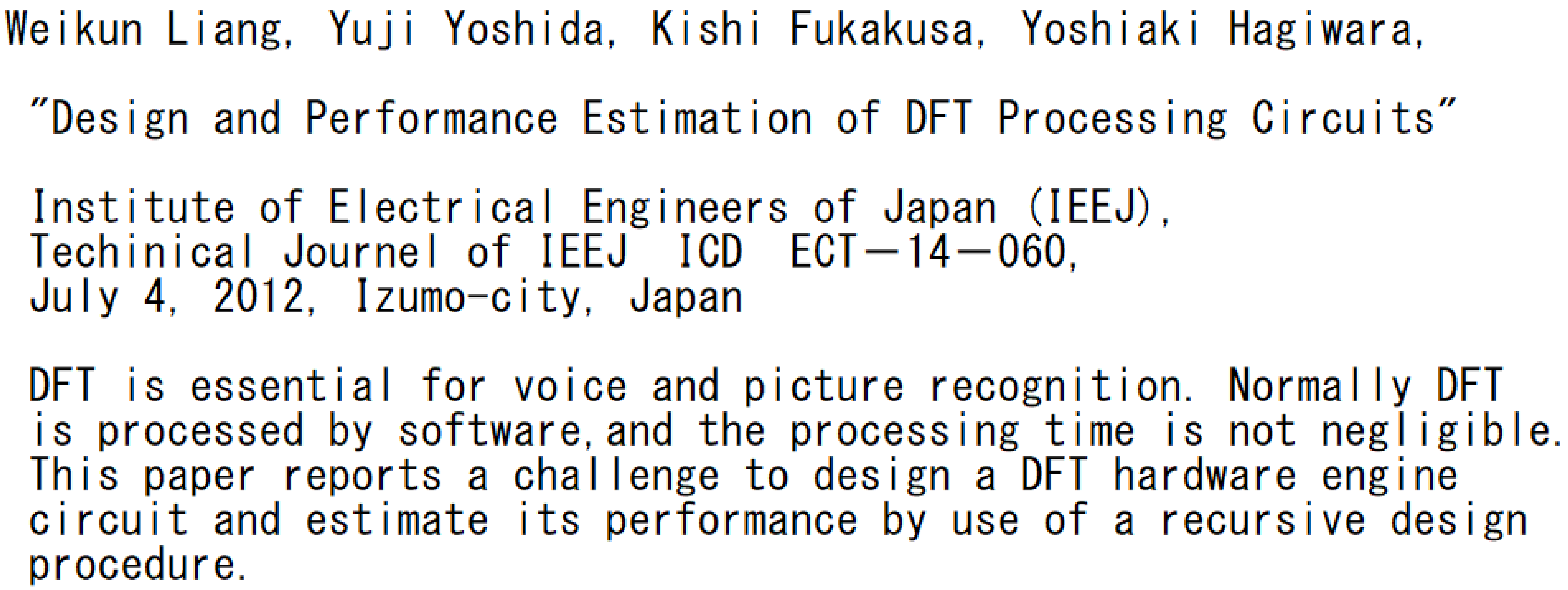
Related published paper on July 4, 2014, Izumo city , Japan

Weikun Liang, Yuji Yoshida, Kishi Fukakusa, Yoshiaki Hagiwara,

″Design and Performance Estimation of DFT Processing Circuits″

Institute of Electrical Engineers of Japan (IEEJ),
Techinical Journel of IEEJ ICD ECT—14—060,
July 4, 2012, Izumo-city, Japan

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme



Weikun Liang, Yuji Yoshida, Kishi Fukakusa, Yoshiaki Hagiwara,

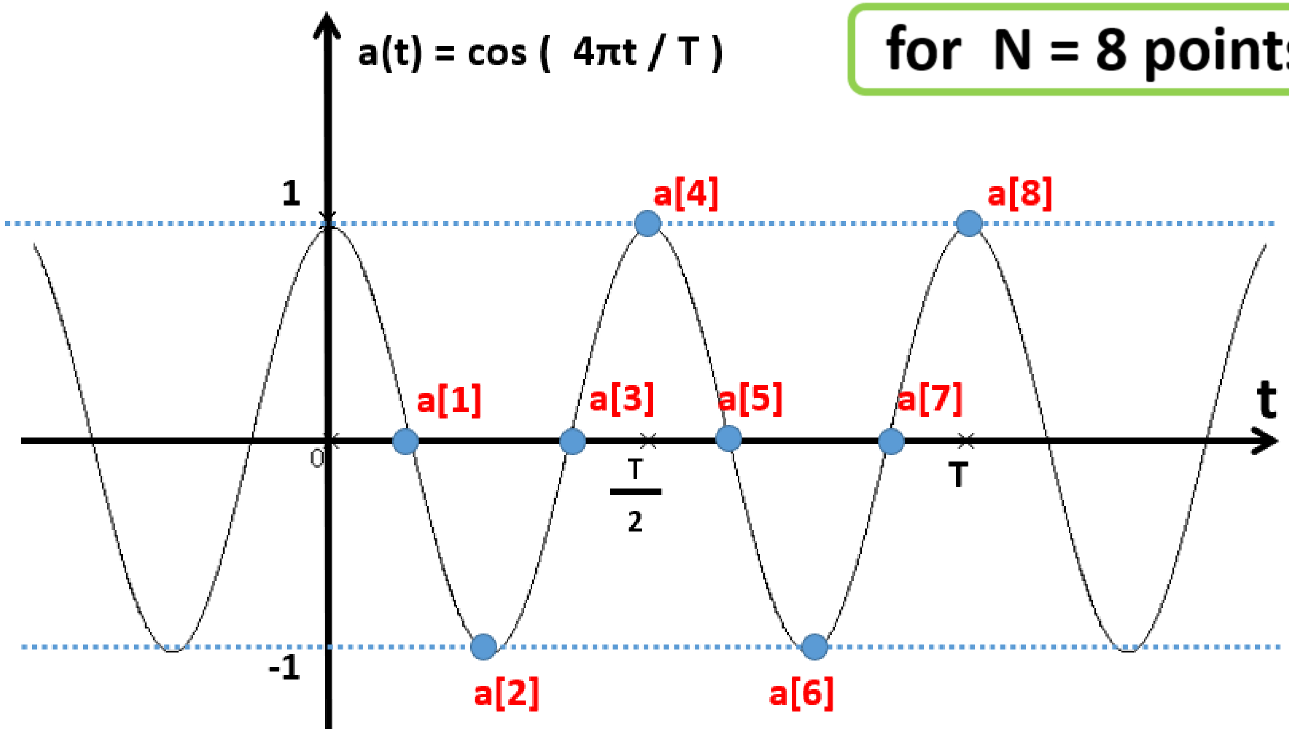
"Design and Performance Estimation of DFT Processing Circuits"

Institute of Electrical Engineers of Japan (IEEJ),
Techinical Journel of IEEJ ICD ECT—14—060,
July 4, 2012, Izumo-city, Japan

DFT is essential for voice and picture recognition. Normally DFT is processed by software, and the processing time is not negligible. This paper reports a challenge to design a DFT hardware engine circuit and estimate its performance by use of a recursive design procedure.

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

Normally DFT is processed by software. However, the processing time is not negligible . To build a powerful AI image sensor system, a DFT digital circuit real time hardware engine is strongly desired.



for N = 8 points

a[ ] ={ a[1], a[2], ······· , a[8] }

A[ ] ={A[1], A[2], ······· , A[8] }

DFT [m][n] = exp (− 2 m n π i / N)

for m = 1, 2, ····· N and n = 1, 2,····· N.

$i^2 = -1$

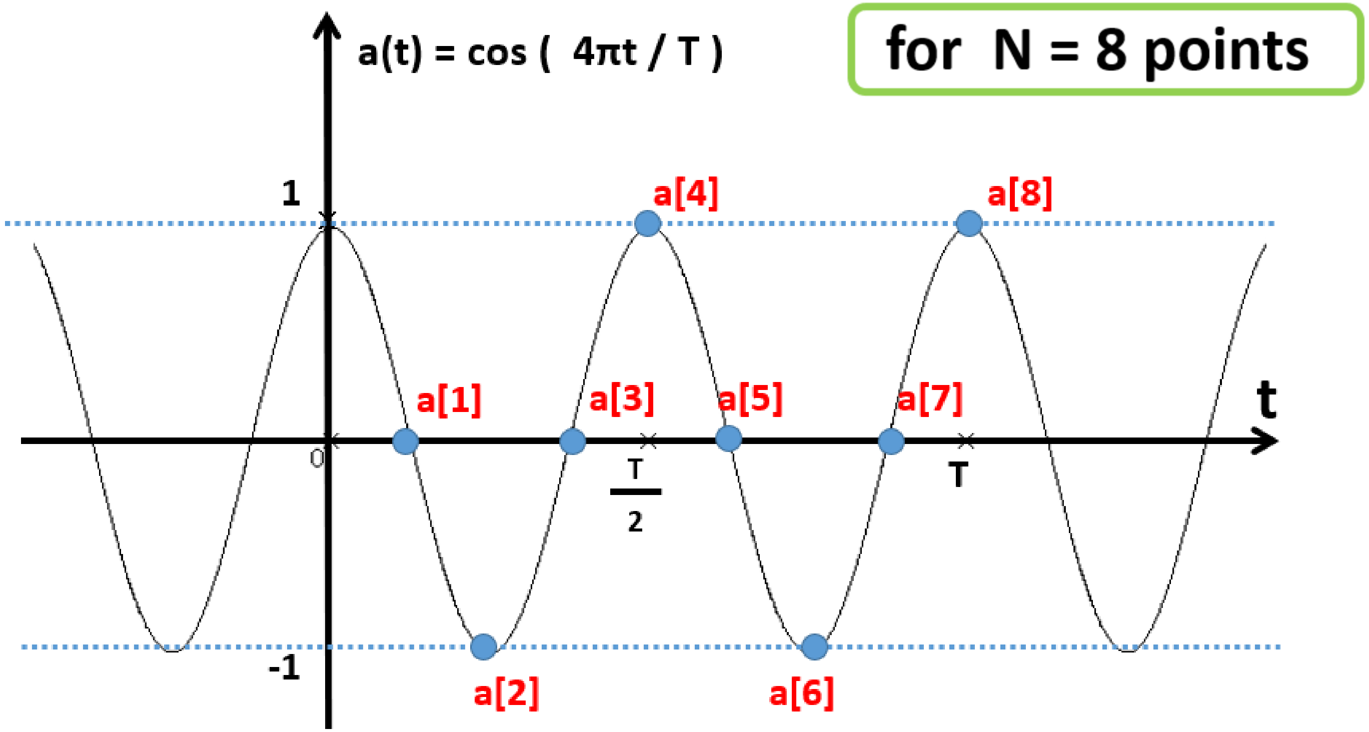
A[ ] = DFT[ ][ ]a[ ]

Sampling Points are at t[k] = kT/8 for k=1 to 8 ;

A simple case of the input signal a(t) = cos (4πt / T) and the N = 8 sampling points

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

The frequency component vector A[ ] can be obtained by multiplying the input signal vector a[ ] by the DFT matrix DFT8[ ][ ] .



for N = 8 points

$$a[\ ] = \{ a[1], a[2], \cdots\cdots , a[8] \}$$

$$A[\ ] = \{A[1], A[2], \cdots\cdots , A[8] \}$$

$$DFT\,[m][n] = \exp\,(- 2\, m\, n\, \pi\, i \,/\, N\,)$$

for m = 1, 2, ····· N and n = 1, 2,····· N.

$$i^2 = -1$$

$$A[\ ] = DFT[\ ][\ ]a[\ ]$$

Sampling Points are at t[k] = kT/8 for k=1 to 8 ;

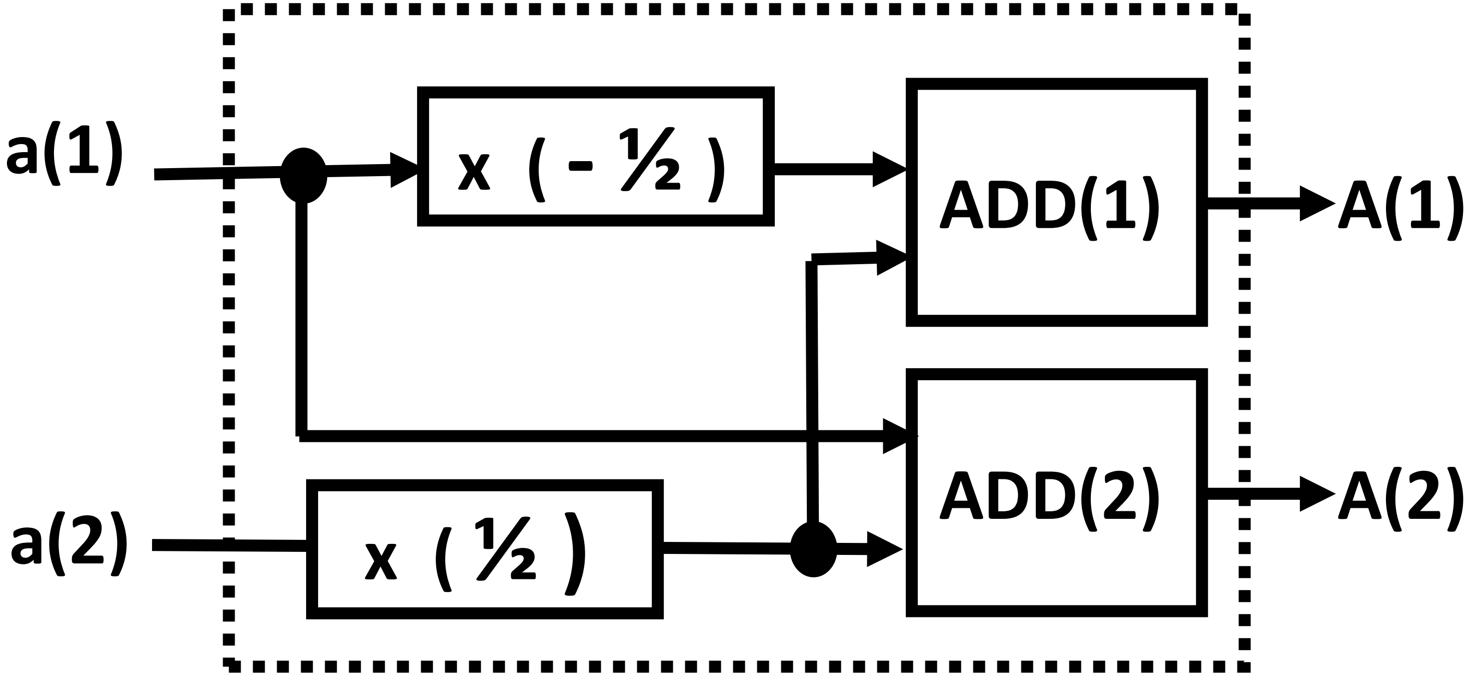
A simple case of the input signal **a(t) = cos (4πt / T)** and the N = 8 sampling points

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

for N = 2 points

DFT2( ) circuit block diagram

ADD( ) is a full adder function (circuit) and
the multiplier function x(½) has the M value of (½).



DFT2( ) circuit function matrix

$$A[\ ] = DFT2[\ ][\ ]\ a[\ ]$$

a[ ] ={ a[1], a[2] }

A[ ] ={A[1], A[2] }

$$j^2 = -1$$

$$DFT2\ [m][n] = \frac{1}{2} \exp\left(- \frac{2\,\pi\ m\ n\ j}{2} \right)$$

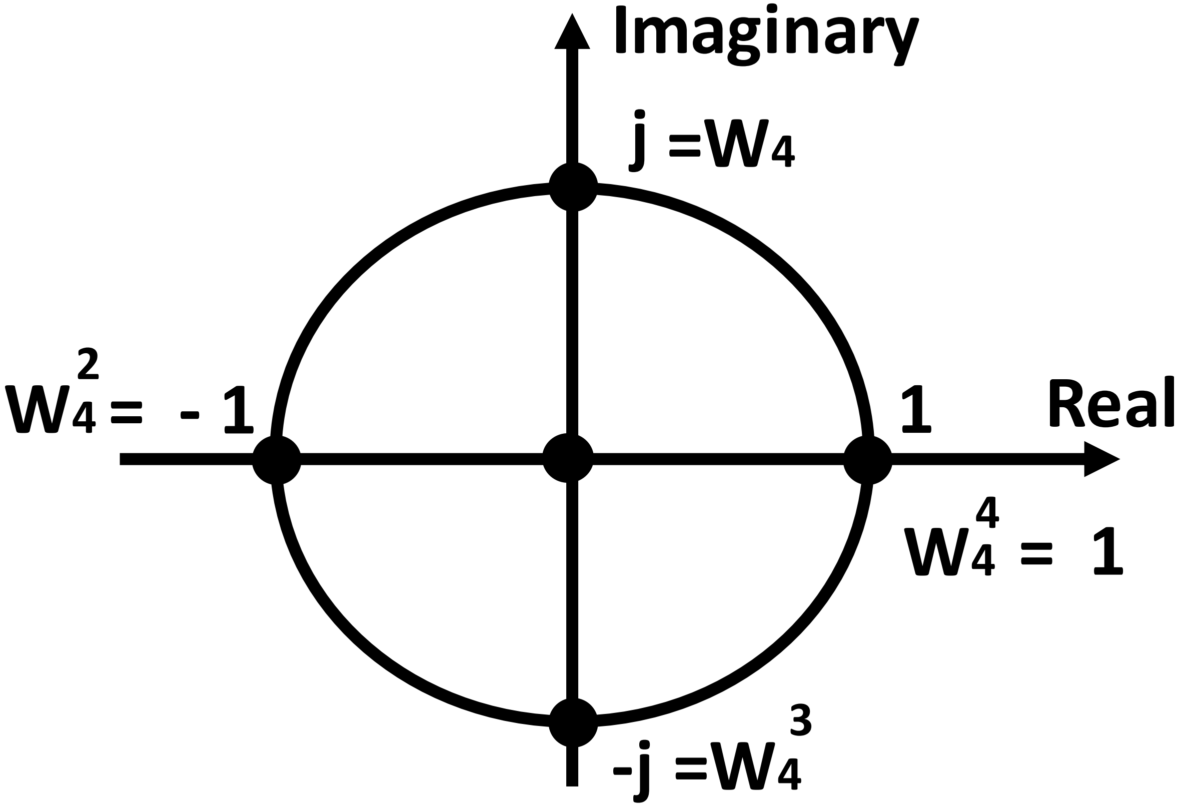
for m = 1, 2 and n = 1,2 .

DFT2( ) Matrix

$$\begin{bmatrix} A(1) \\ A(2) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} a(1) \\ a(2) \end{bmatrix}$$

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

for N = 4 points



$$\text{DFT4}\,[m][n] = \frac{1}{4}\exp\left(-\frac{2\pi\, m\, n\, j}{4}\right)$$

for m = 1, 2 ,3,4 and n = 1,2,3,4 .

DFT4( ) circuit function matrix

A[ ] = DFT4[ ][ ] a[ ]

a[ ] ={ a[1], a[2],a[3].a[4] }

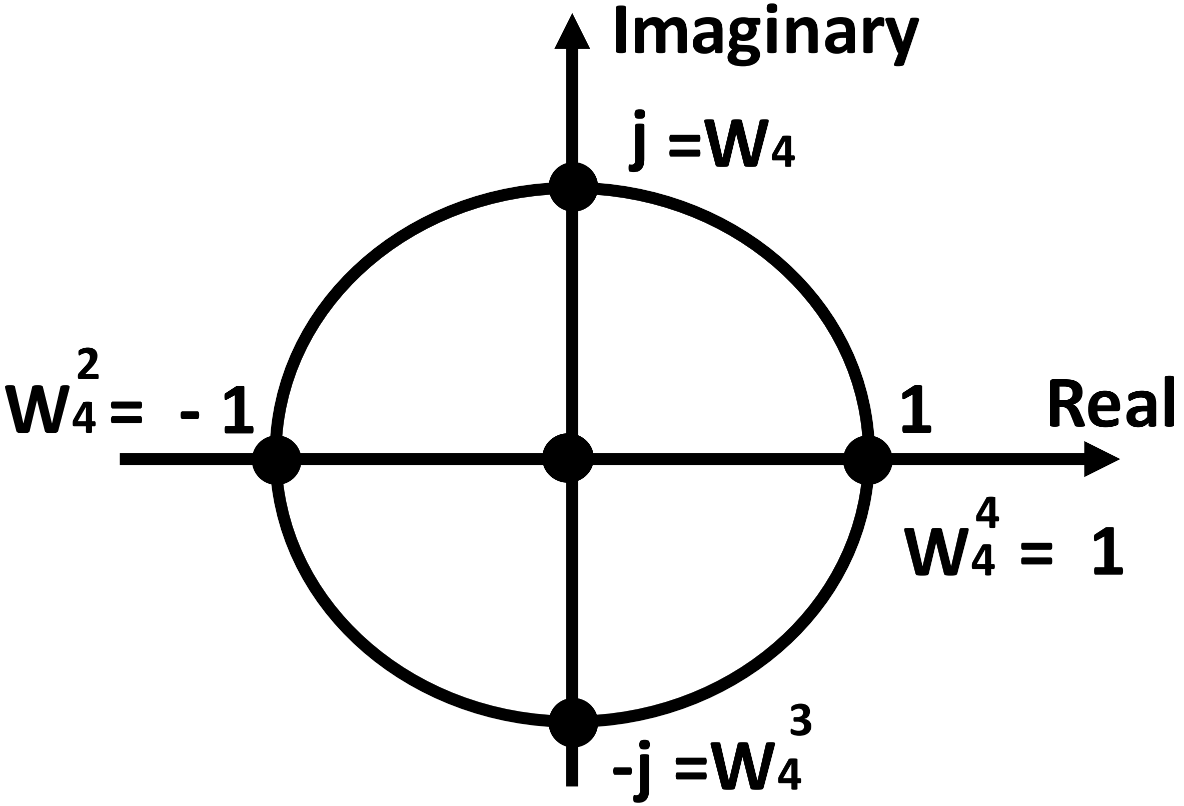
A[ ] ={A[1], A[2].A[3],A[4] }

$$\begin{vmatrix} A(1) \\ A(2) \\ A(3) \\ A(4) \end{vmatrix} = \frac{1}{4} \begin{bmatrix} -j & -1 & j & 1 \\ -1 & 1 & -1 & 1 \\ j & -1 & -j & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{vmatrix} a(1) \\ a(2) \\ a(3) \\ a(4) \end{vmatrix}$$

DFT4( ) Matrix

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

for N = 4 points



$$DFT4\,[m][n] = \frac{1}{4}\exp\left(-\frac{2\,\pi\, m\, n\, j}{4}\right)$$

for m = 1, 2 ,3,4 and n = 1,2,3,4 .

DFT4( ) Matrix

$$\begin{bmatrix} A(2) \\ A(4) \end{bmatrix} = \frac{1}{4}\underbrace{\begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}}_{DFT2(1)}\begin{bmatrix} a(1) \\ a(2) \end{bmatrix} + \frac{1}{4}\underbrace{\begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}}_{DFT2(2)}\begin{bmatrix} a(3) \\ a(4) \end{bmatrix}$$

$$\begin{bmatrix} A(1) \\ A(3) \end{bmatrix} = \frac{1}{4}\underbrace{\begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}}_{DFT2(3)}\begin{bmatrix} -j & \{ -a(1) + a(3) \} \\ & \{ -a(2) + a(4) \} \end{bmatrix}$$

$$\left|\begin{matrix} A(1) \\ A(2) \\ A(3) \\ A(4) \end{matrix}\right| = \frac{1}{4}\begin{array}{|c|c|c|c|} \hline -j & -1 & j & 1 \\ \hline -1 & 1 & -1 & 1 \\ \hline j & -1 & -j & 1 \\ \hline 1 & 1 & 1 & 1 \\ \hline \end{array}\left|\begin{matrix} a(1) \\ a(2) \\ a(3) \\ a(4) \end{matrix}\right|$$

DFT4( ) Matrix

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

for N = 4 points

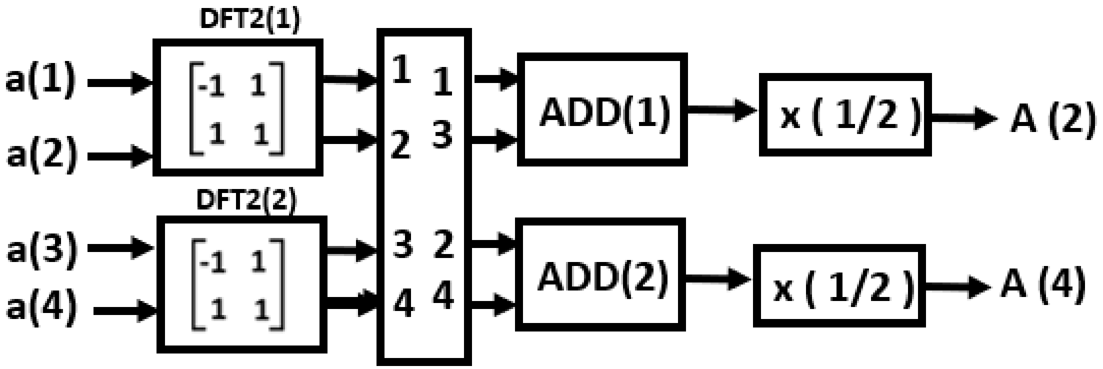
DFT4( ) Matrix

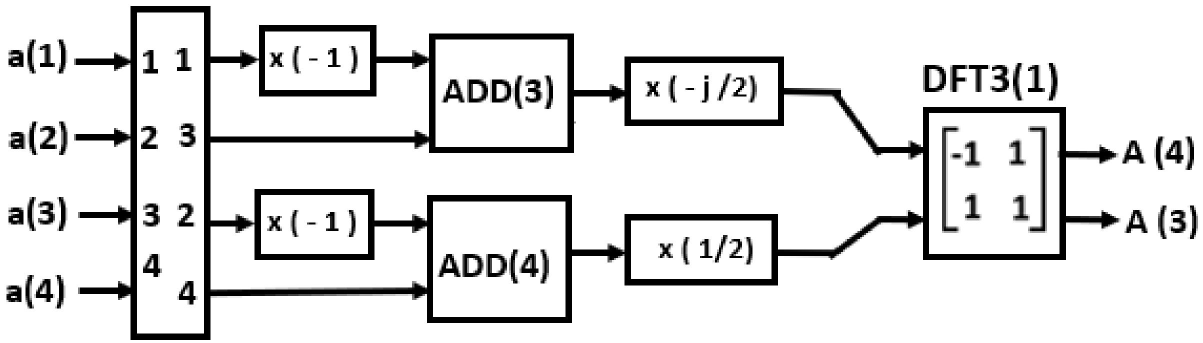
$$\left|\begin{matrix} A(1) \\ A(2) \\ A(3) \\ A(4) \end{matrix}\right| = \frac{1}{4} \begin{array}{|c|c|c|c|} \hline -j & -1 & j & 1 \\ \hline -1 & 1 & -1 & 1 \\ \hline j & -1 & -j & 1 \\ \hline 1 & 1 & 1 & 1 \\ \hline \end{array} \left|\begin{matrix} a(1) \\ a(2) \\ a(3) \\ a(4) \end{matrix}\right|$$

DFT4( ) circuit block diagram

$$\begin{bmatrix} A(2) \\ A(4) \end{bmatrix} = \frac{1}{4} \underset{}{\overset{\text{DFT2(1)}}{\begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}}} \begin{bmatrix} a(1) \\ a(2) \end{bmatrix} + \frac{1}{4} \overset{\text{DFT2(2)}}{\begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}} \begin{bmatrix} a(3) \\ a(4) \end{bmatrix}$$

$$\begin{bmatrix} A(1) \\ A(3) \end{bmatrix} = \frac{1}{4} \overset{\text{DFT2(3)}}{\begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}} \begin{bmatrix} -j \ \{ -a(1) + a(3) \} \\ \{ -a(2) + a(4) \} \end{bmatrix}$$





DFT2( ) is a 2 bit DFT circuit and x(-j)( ) and x(¼)( ) are multipliers with the M values of (-j) and (¼) respectively .

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

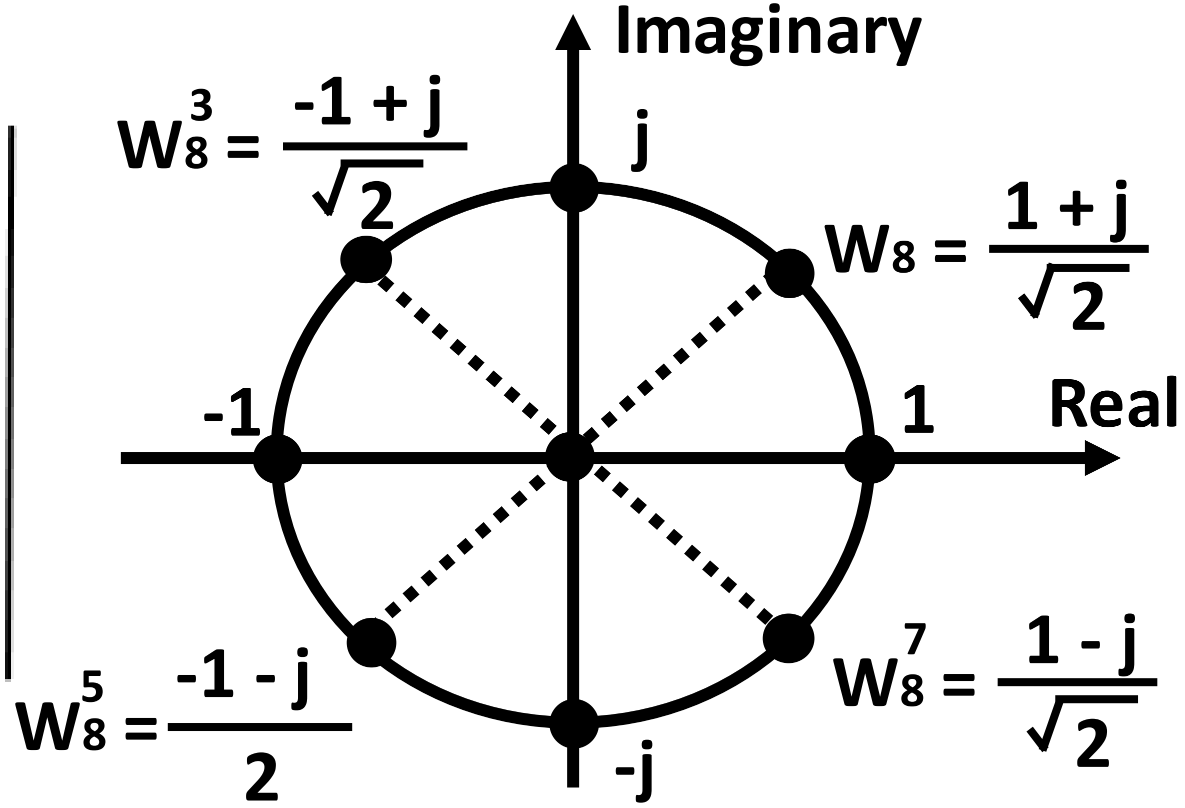
DFT8( ) circuit block diagram

for N = 8 points

A[ ] = DFT8[ ][ ] a[ ]

A[ ] = DFT8[ ][ ] a[ ] matrix for N = 8

$$\begin{vmatrix} A(1) \\ A(2) \\ A(3) \\ A(4) \\ A(5) \\ A(6) \\ A(7) \\ A(8) \end{vmatrix} = \frac{1}{8} \begin{array}{|c|c|c|c|c|c|c|c|} \hline W_8^7 & -j & W_8^5 & -1 & W_8^3 & j & W_8 & 1 \\ \hline -j & -1 & j & 1 & -j & -1 & j & 1 \\ \hline W_8^5 & j & W_8^7 & -1 & W_8 & -j & W_8^3 & 1 \\ \hline -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ \hline W_8^3 & -j & W_8 & -1 & W_8^7 & j & W_8^5 & 1 \\ \hline j & -1 & -j & 1 & j & -1 & -j & 1 \\ \hline W_8 & j & W_8^3 & -1 & W_8^5 & -j & W_8^7 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline \end{array} \begin{vmatrix} a(1) \\ a(2) \\ a(3) \\ a(4) \\ a(5) \\ a(6) \\ a(7) \\ a(8) \end{vmatrix}$$


Imaginary
Real
j
-j
-1
1
$W_8^3 = \frac{-1 + j}{\sqrt{2}}$
$W_8 = \frac{1 + j}{\sqrt{2}}$
$W_8^5 = \frac{-1 - j}{2}$
$W_8^7 = \frac{1 - j}{\sqrt{2}}$

a[ ] ={ a[1], a[2],a[3].a[4], a[5], a[6],a[7].a[8] }

A[ ] ={A[1], A[2].A[3],A[4], A[5], A[6].A[7],A[8]}

$$\text{DFT }[m][n] = \frac{1}{8} \exp\left(- \frac{2 \pi m n j}{8} \right)$$

for m = 1, 2,3,4 ,,,, 8 and n = 1,2,3,4, ,,,,,8

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

DFT8( ) circuit block diagram

for N = 8 points

$$A[\ ] = DFT8[\ ][\ ]\ a[\ ]$$

$$\left|\begin{matrix} A(1) \\ A(3) \\ A(5) \\ A(7) \end{matrix}\right| = \frac{1}{8} \begin{array}{r|c|c|c|c|} \cline{2-5} (\ W_8\ )\ x & -j & j & -1 & 1 \\ \cline{2-5} (\ W_8^5\ )\ x & -j & j & -1 & 1 \\ \cline{2-5} (\ W_8^3\ )\ x & j & -j & -1 & 1 \\ \cline{2-5} (\ W_8^7\ )\ x & j & -j & -1 & 1 \\ \cline{2-5} \end{array} \left|\begin{matrix} a(1) \\ a(3) \\ a(5) \\ a(7) \end{matrix}\right| + \frac{1}{8} \begin{array}{|c|c|c|c|} \hline -j & j & -1 & 1 \\ \hline -j & j & -1 & 1 \\ \hline j & -j & -1 & 1 \\ \hline j & -j & -1 & 1 \\ \hline \end{array} \left|\begin{matrix} a(2) \\ a(4) \\ a(6) \\ a(8) \end{matrix}\right|$$

DFT4( ) Matrix

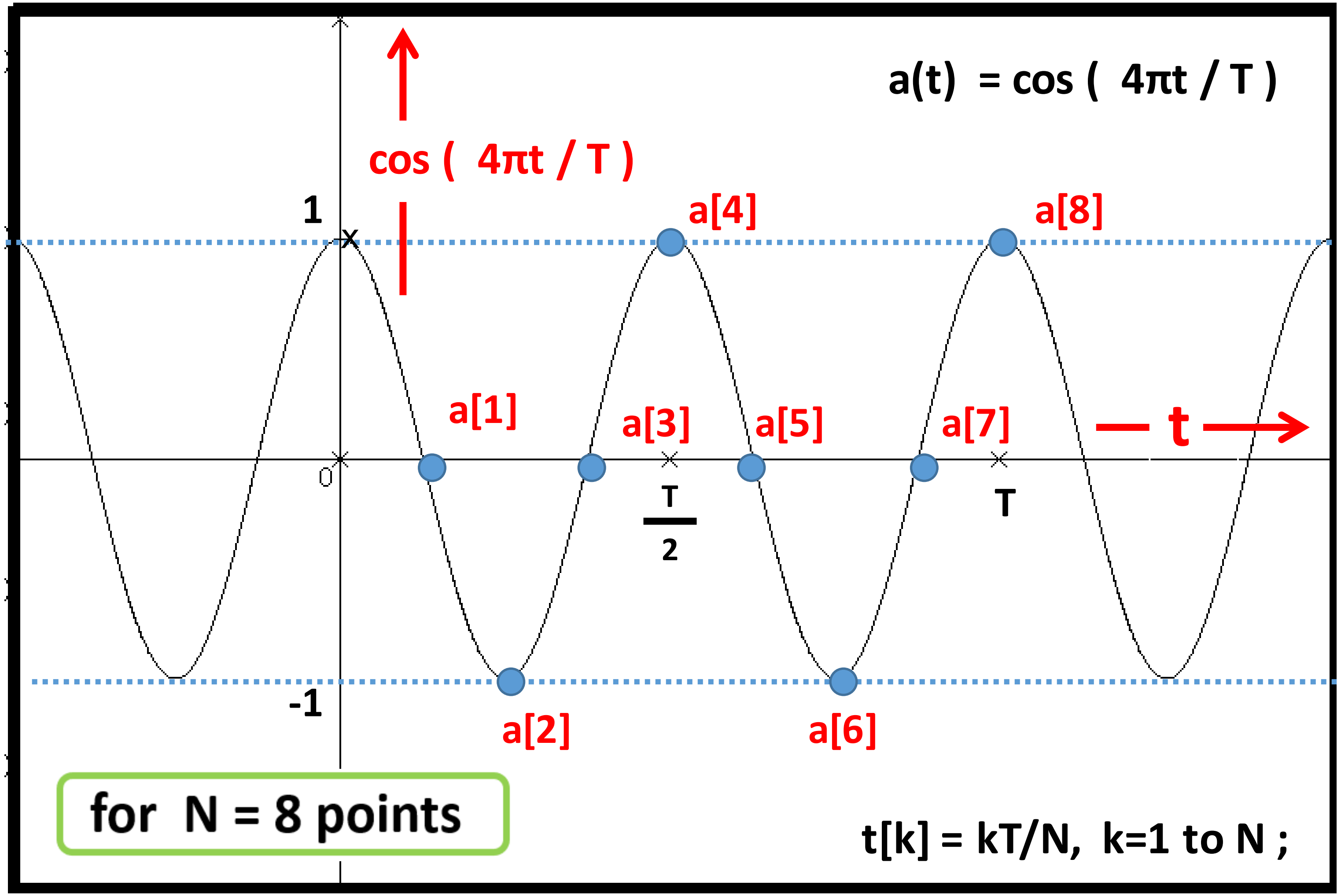
DFT4( ) Matrix

$$\left|\begin{matrix} A(2) \\ A(4) \\ A(6) \\ A(8) \end{matrix}\right| = \frac{1}{8} \begin{array}{|c|c|c|c|} \hline -j & -1 & j & 1 \\ \hline -1 & 1 & -1 & 1 \\ \hline j & -1 & -j & 1 \\ \hline 1 & 1 & 1 & 1 \\ \hline \end{array} \left|\begin{matrix} a(1) \\ a(2) \\ a(3) \\ a(4) \end{matrix}\right| + \frac{1}{8} \begin{array}{|c|c|c|c|} \hline -j & -1 & j & 1 \\ \hline -1 & 1 & -1 & 1 \\ \hline j & -1 & -j & 1 \\ \hline 1 & 1 & 1 & 1 \\ \hline \end{array} \left|\begin{matrix} a(4) \\ a(5) \\ a(6) \\ a(7) \end{matrix}\right|$$

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme

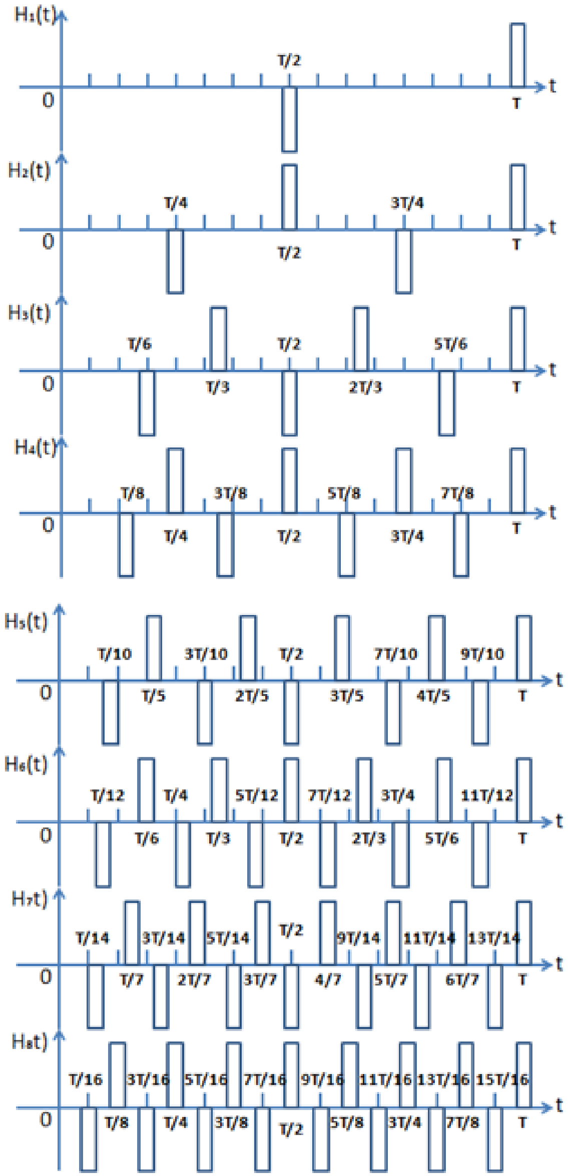
a[1]=	0
a[2]=	-1
a[3]=	0
a[4]=	1
a[5]=	0
a[6]=	-1
a[7]=	0
a[8]=	1

Ar[1]=	0
Ar[2]=	0.5
Ar[3]=	0
Ar[4]=	0
Ar[5]=	0
Ar[6]=	0
Ar[7]=	0.5
Ar[8]=	0



$$a(t) \;=\; \sum_{h=1}^{8} A[h]\exp(2h\pi tj)\;; \qquad A[m] = \sum_{k=1}^{8} (1/8)\,\mathrm{DFT8}\,[m][k]\,a[k]\;;$$

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme



Conventional DFT Case

$$a(t) \;=\; \sum_{m=1}^{N} h_m(t)\, A[m]$$

From k=1 to N , t[n]= kT/N and

$$a[k] = a(kT/N) \;=\; \sum_{m=1}^{N} h_m(kT/N)\; A[m]$$

$$H[n][m] = h_m(nT/N)$$

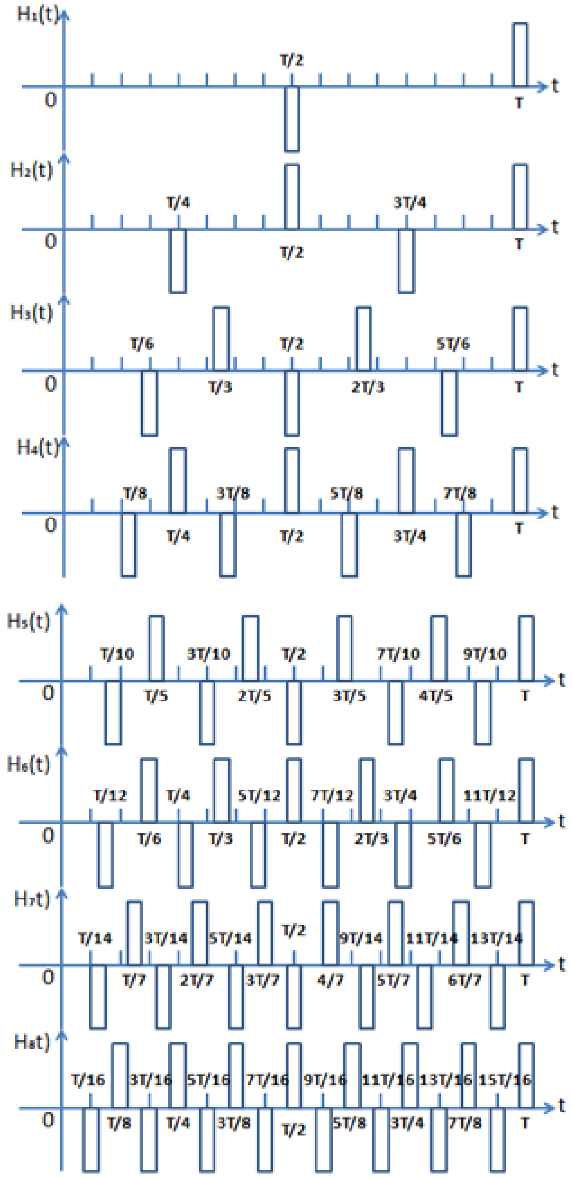
Matrix H[ ][ ] with each element value H[n][m],

$$a[n] = \sum_{m=1}^{N} H[n][m]\, A[m] \quad ; \qquad a(t) = \sum_{m=1}^{N} h_m(t)\, A[m]\;;$$

a[ ] = H[ ] [ ] A[ ]

for N = 8 points

Conventional Uniform Interval Δ(T/N) DFT ( ) Sampling Scheme



Conventional DFT Matrix

$$
\mathrm{DFT8}\,[\;]\,[\;] \;=\; \begin{array}{c|cccccccc|}
 & \multicolumn{8}{c}{\xrightarrow{\qquad\qquad m}} \\
\downarrow & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\
\downarrow & 0 & -1 & 0 & 1 & 0 & -1 & 0 & 1 \\
\downarrow & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\
\downarrow & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
\downarrow & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\
\downarrow & 0 & -1 & 0 & 1 & 0 & -1 & 0 & 1 \\
n & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\
 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{array}
$$

$H[n][m] = h_m\ (\,nT/N\,)$

Matrix H[ ][ ] has each element H[n][m].

$$a[n] = \sum_{m=1}^{N} H[n][m]\, f[\,m\,] \quad ; \quad a(\,t\,) \;=\; \sum_{m=1}^{N} h_m(\,t\,)\, f[\,m\,]$$

for N = 8 points

a[ ] = H[ ] [ ] A[ ]

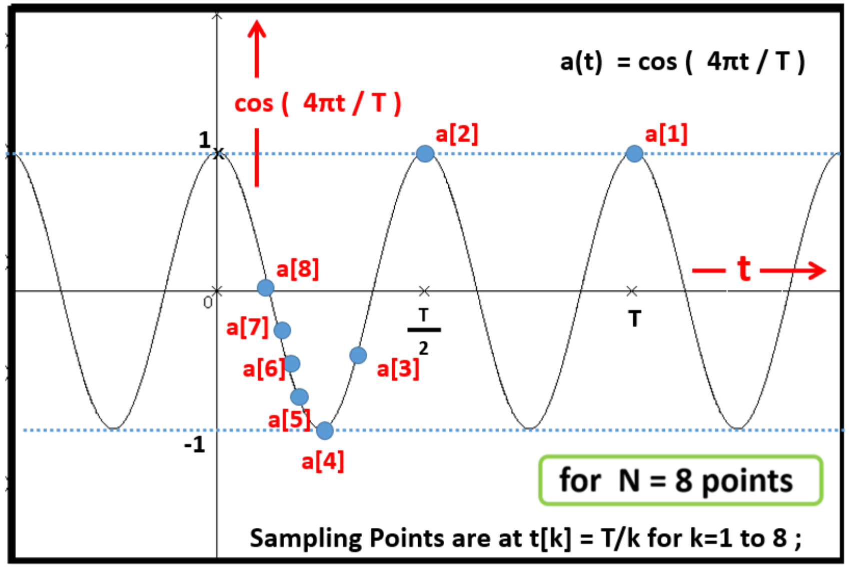
9 Artificial Intelligent Image Sensor

Artificial Intelligent Image Sensor needs many kinds of powerful hardware real time engines. Nonuniform T/k Sampling Scheme with the Fast H [ ][ ] Transformation Matrix is also one of them.

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

a[1]= 1
a[2]= 1
a[3]= - 0.5
a[4]= - 1
a[5]= - 0.809
a[6]= - 0.5
a[7]= - 0.223
a[8]= 0

A[1]= 1.032
A[2]= 1.5
A[3]= 0
A[4]= 0
A[5]= - 0.809
A[6]= - 0.5
A[7]= - 0.223
A[8]= 0



Sampling Points are at t[k] = T/k for k=1 to 8 ;

Compute $A[m] = \sum_{k=1}^{8} (1/8)\, H[m][k]\, a[k]$; Then we have $a(t) = \sum_{m=1}^{8} A[m]\, H_m(t)$;

Step(1) Define $H_m(t)$ functions

Step(2) Define H[m][k]

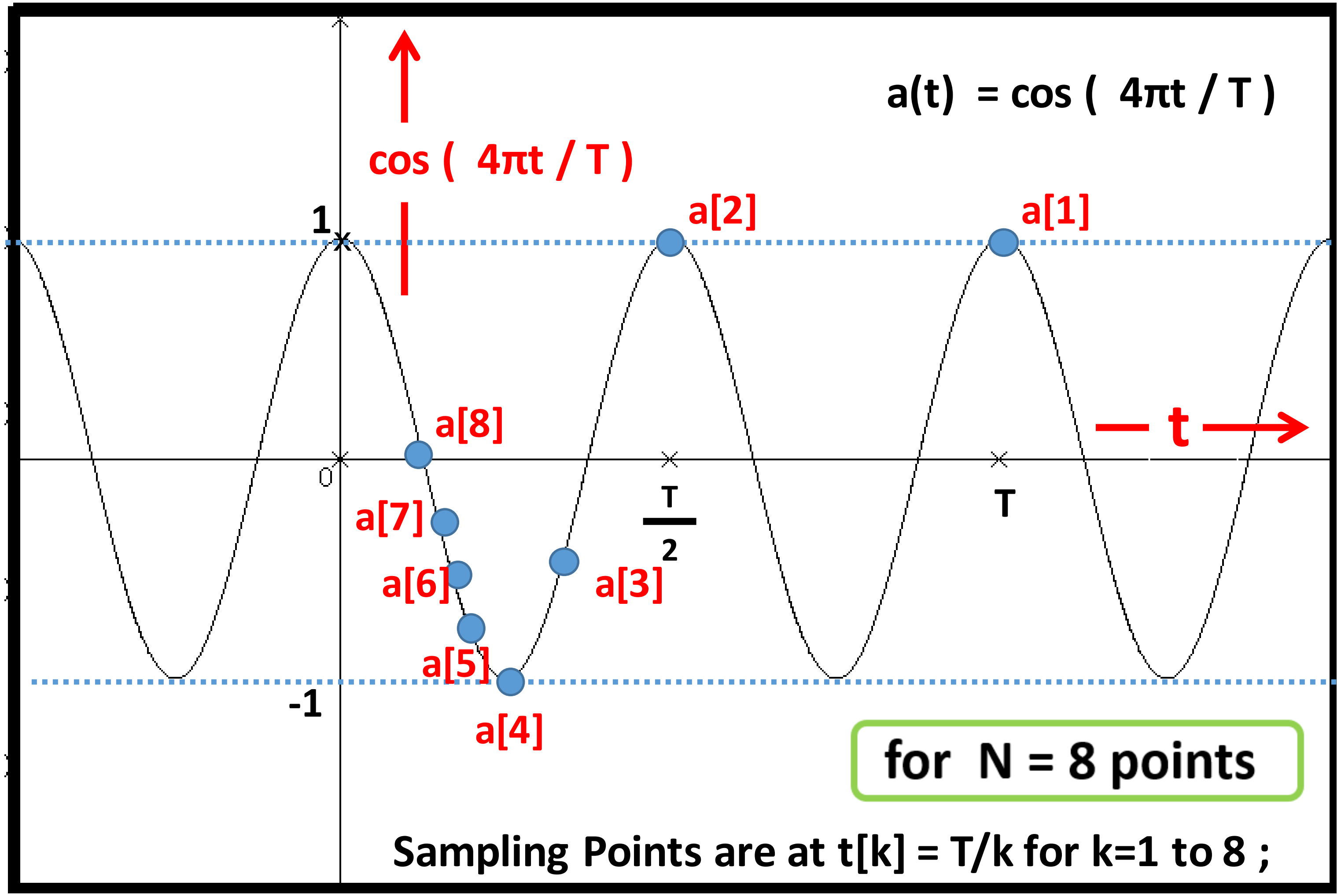
Step(3) Obtain a[m]

Step(4) Obtain A[m]

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

a[1]= 1
a[2]= 1
a[3]= - 0.5
a[4]= - 1
a[5]= - 0.809
a[6]= - 0.5
a[7]= - 0.223
a[8]= 0

A[1]= 1.032
A[2]= 1.5
A[3]= 0
A[4]= 0
A[5]= - 0.809
A[6]= - 0.5
A[7]= - 0.223
A[8]= 0



for N = 8 points

Sampling Points are at t[k] = T/k for k=1 to 8 ;

Compute $A[m] = \sum_{k=1}^{8} (1/8)\, H[m][k]\, a[k]$; Then we have $a(t) = \sum_{m=1}^{8} A[m]\, H_m(t)$;

Nonuniform T/k Sampling Scheme
with Fast H [ ][ ] Transformation Matrix

Design and Performance Estimation of DFT Processing Circuits

Japanese Patent Application JPA 2014-135497

filed on July 1, 2014 by Yoshiaki Hagiwara.

"The digital circuit engine design to convert the time
domain information to the frequency domain information"

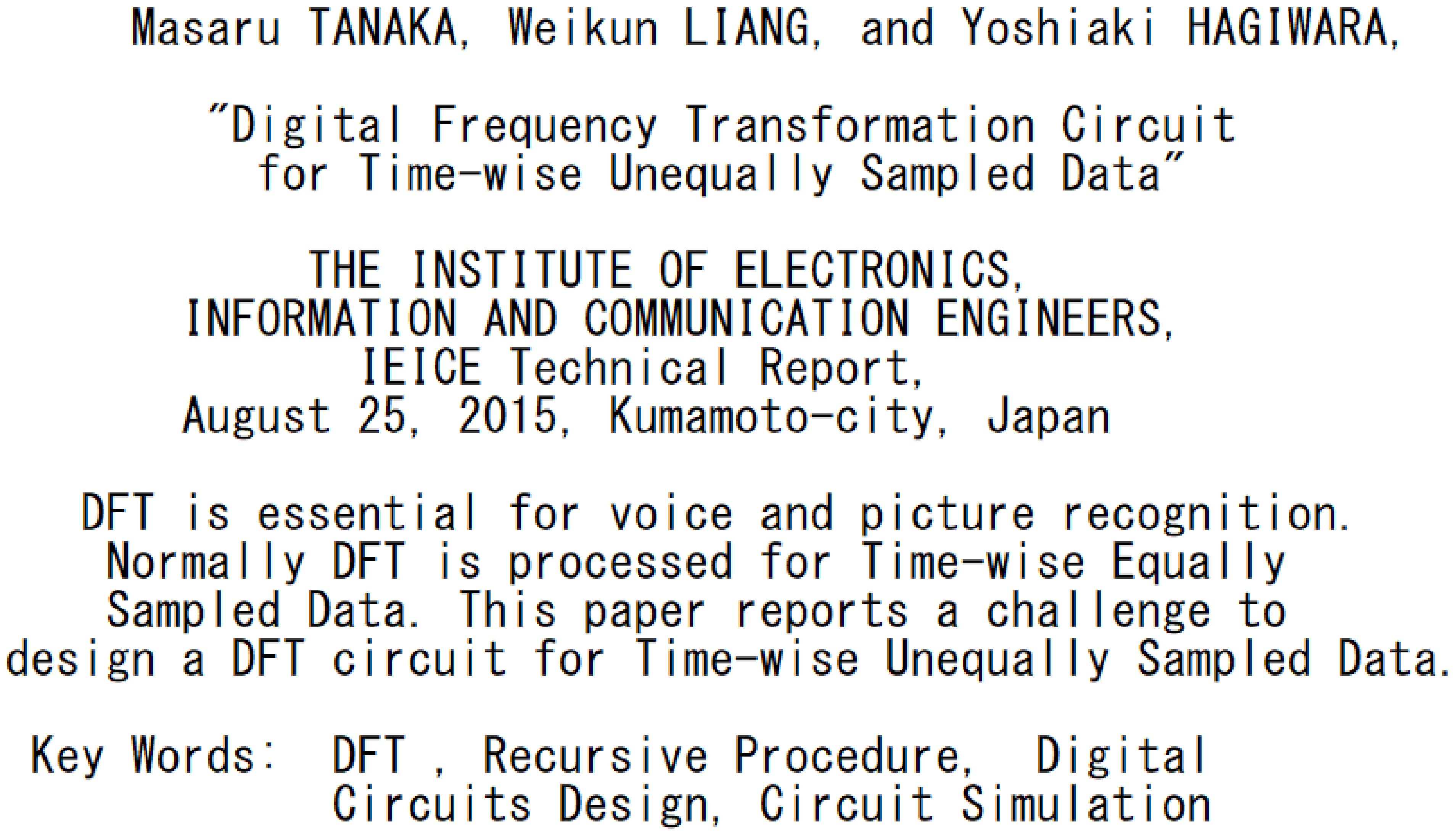
Related published paper on

Masaru TANAKA, Weikun LIANG, and Yoshiaki HAGIWARA,

"Digital Frequency Transformation Circuit
for Time-wise Unequally Sampled Data"

THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS,
IEICE Technical Report,
August 25, 2015, Kumamoto-city, Japan

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix



Masaru TANAKA, Weikun LIANG, and Yoshiaki HAGIWARA,

"Digital Frequency Transformation Circuit
for Time-wise Unequally Sampled Data"

THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS,
IEICE Technical Report,
August 25, 2015, Kumamoto-city, Japan

DFT is essential for voice and picture recognition. Normally DFT is processed for Time-wise Equally Sampled Data. This paper reports a challenge to design a DFT circuit for Time-wise Unequally Sampled Data.

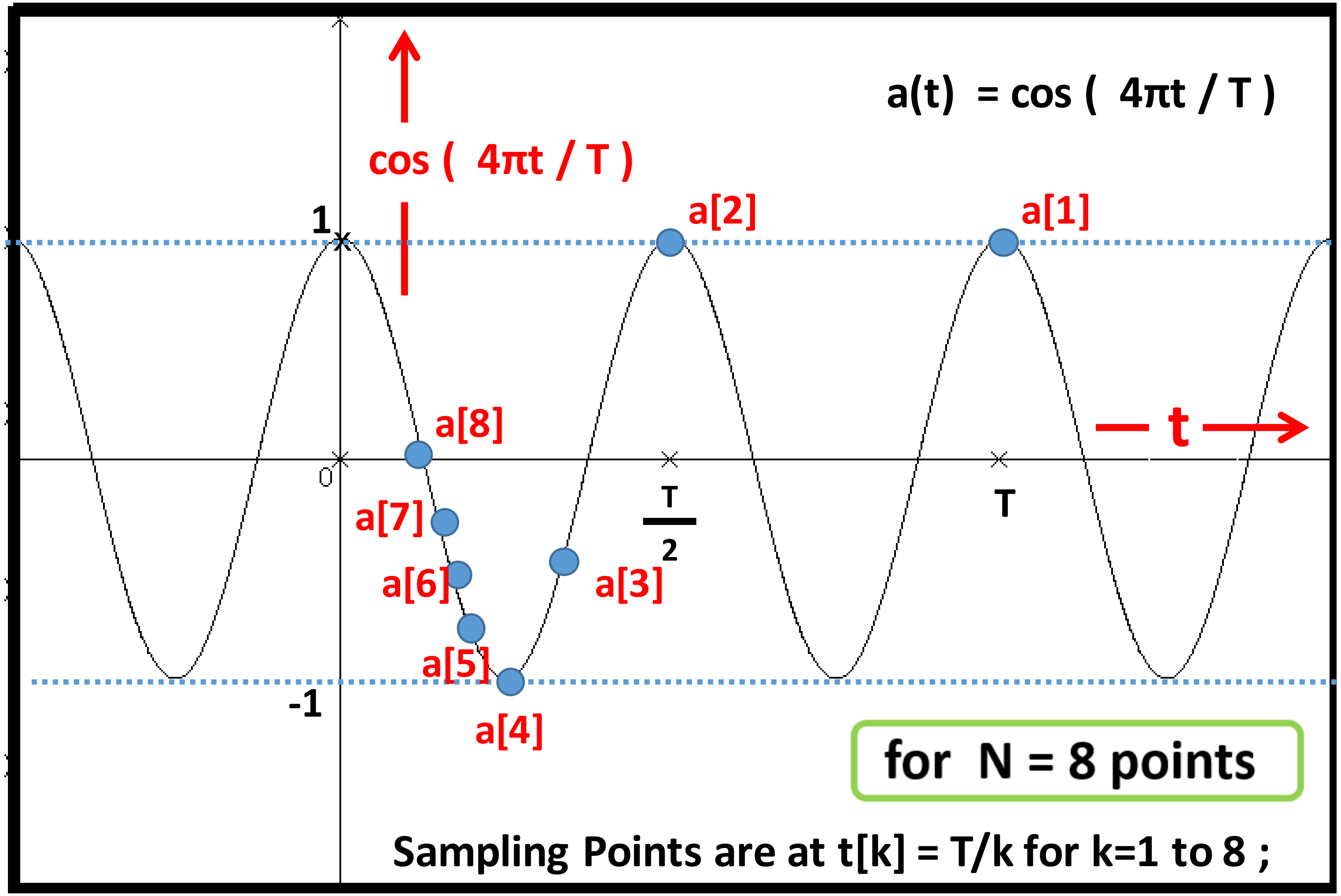
Key Words: DFT , Recursive Procedure, Digital
Circuits Design, Circuit Simulation

Step(1) Define $H_m(t)$ functions

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

a[1]= 1
a[2]= 1
a[3]= - 0.5
a[4]= - 1
a[5]= - 0.809
a[6]= - 0.5
a[7]= - 0.223
a[8]= 0

A[1]= 1.032
A[2]= 1.5
A[3]= 0
A[4]= 0
A[5]= - 0.809
A[6]= - 0.5
A[7]= - 0.223
A[8]= 0



for N = 8 points

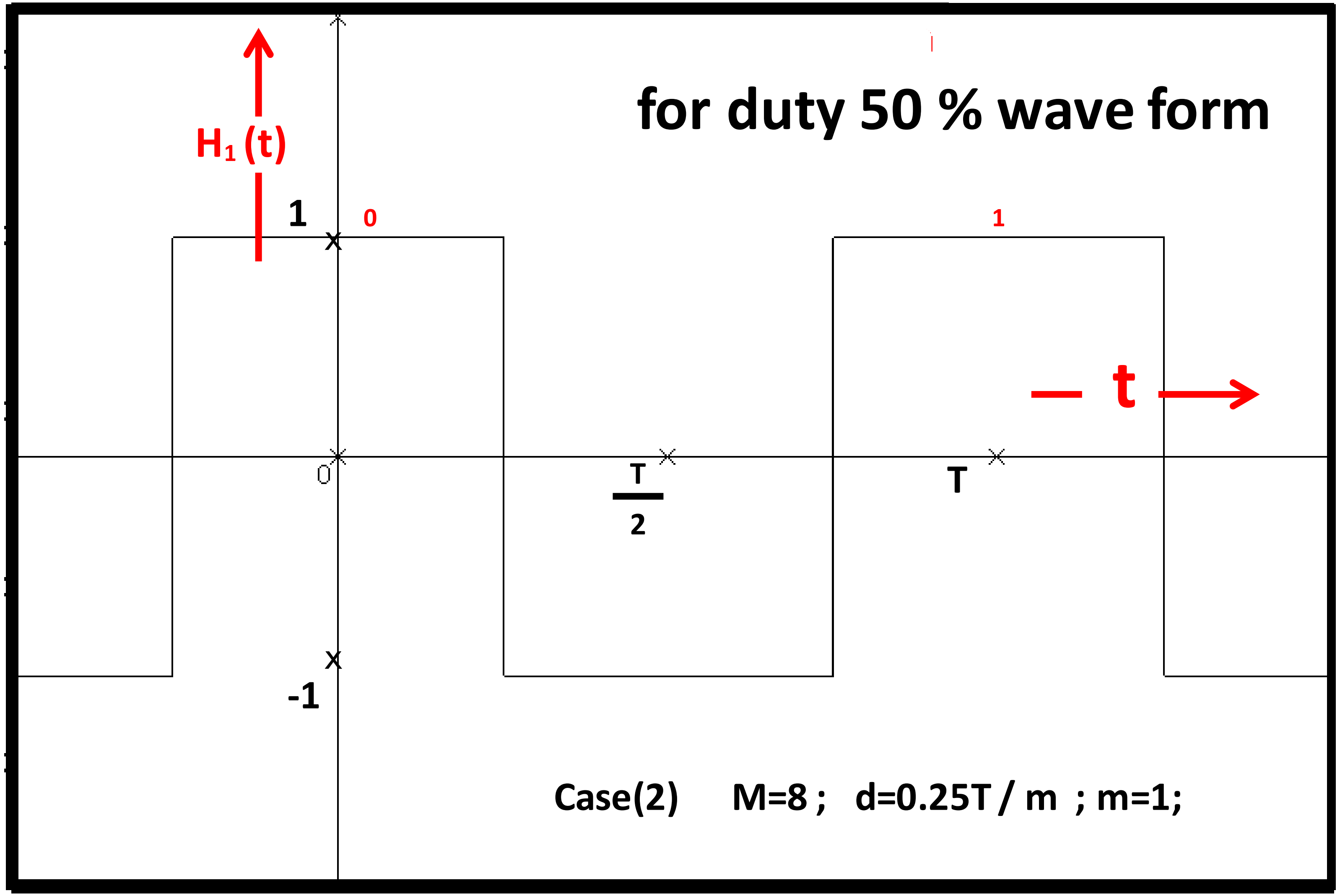
Sampling Points are at t[k] = T/k for k=1 to 8 ;

Compute $A[m] = \sum_{k=1}^{8} (1/8)\, H[m][k]\, a[k]$; Then we have $a(t) = \sum_{m=1}^{8} A[m]\, H_m(t)$;

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 1;

for N = 8 points



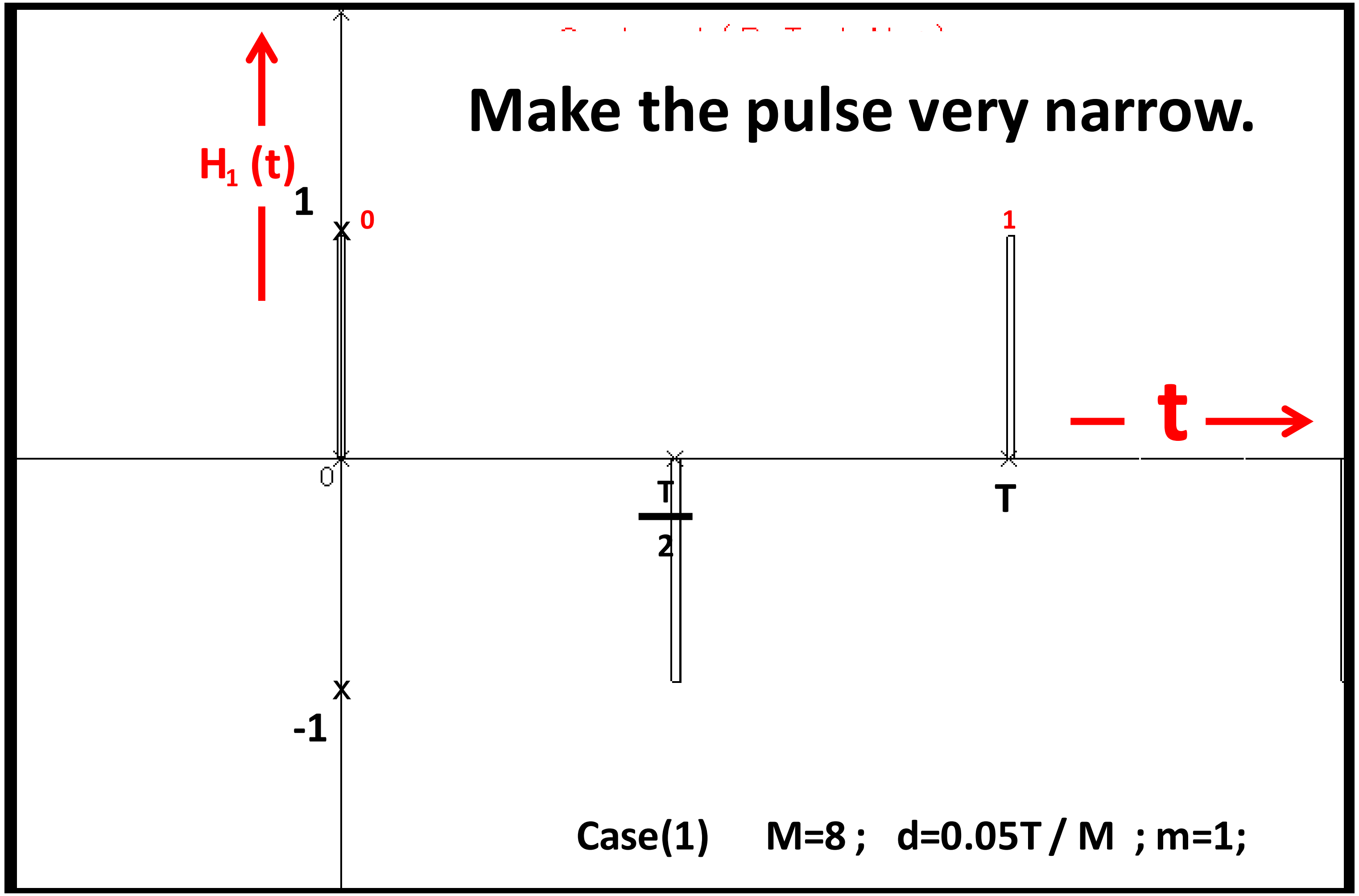
for duty 50 % wave form

Case(2) M=8 ; d=0.25T / m ; m=1;

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 1;

for N = 8 points



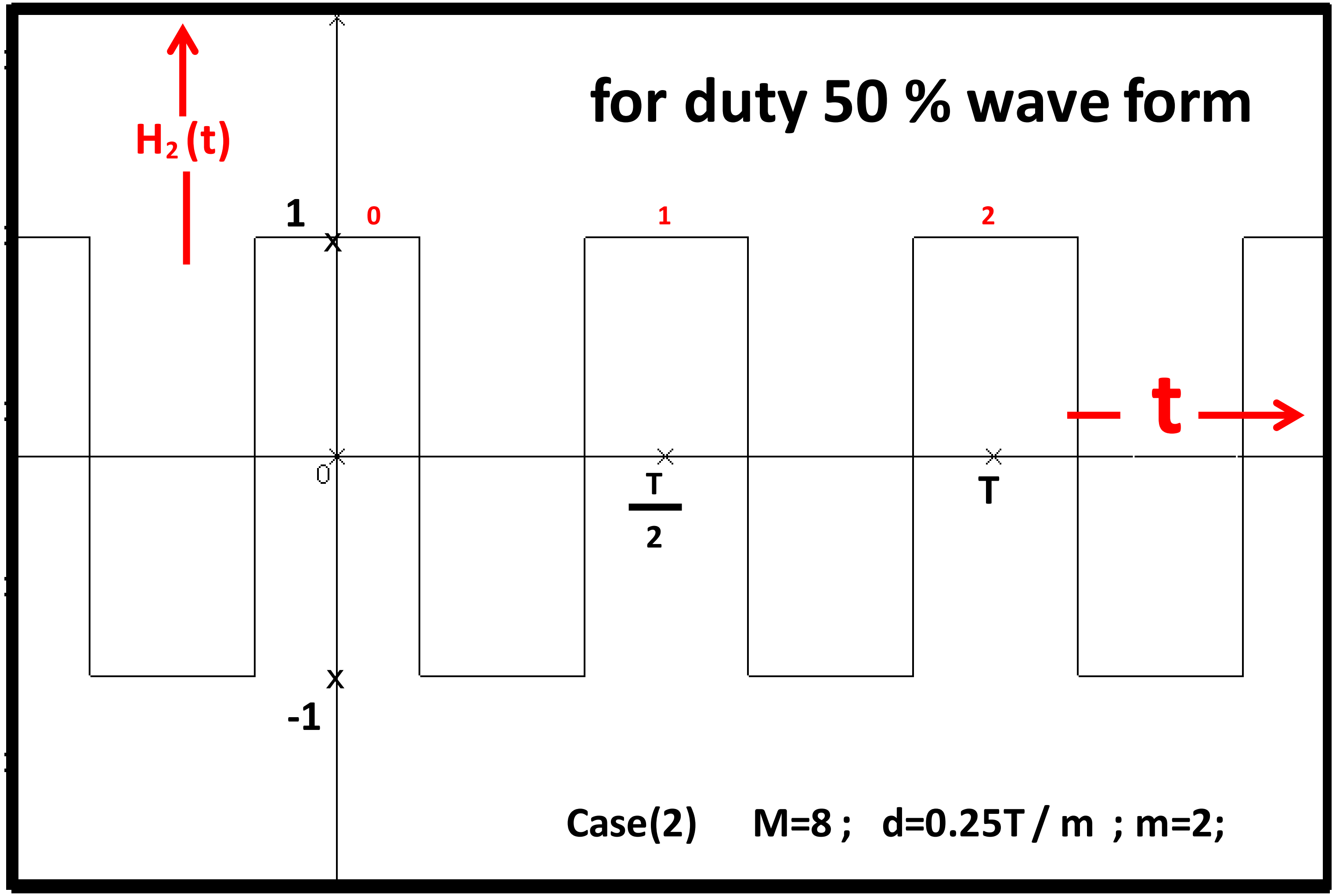
Make the pulse very narrow.

Case(1) M=8 ; d=0.05T / M ; m=1;

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Define H_m (t) function ; m = 2;

for N = 8 points

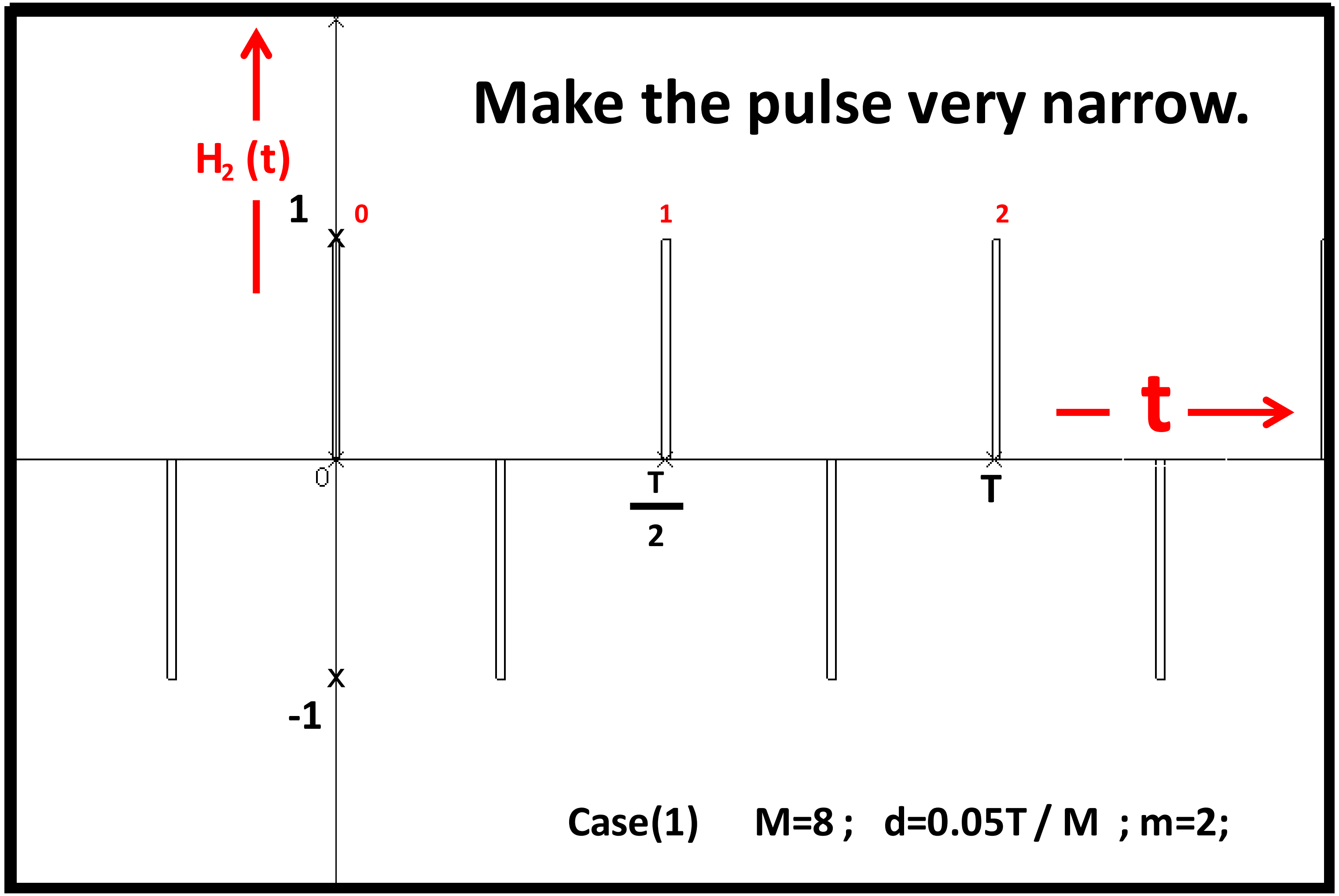


Case(2) M=8 ; d=0.25T / m ; m=2;

Nonuniform T/k Sampling Scheme
with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 2;

for N = 8 points



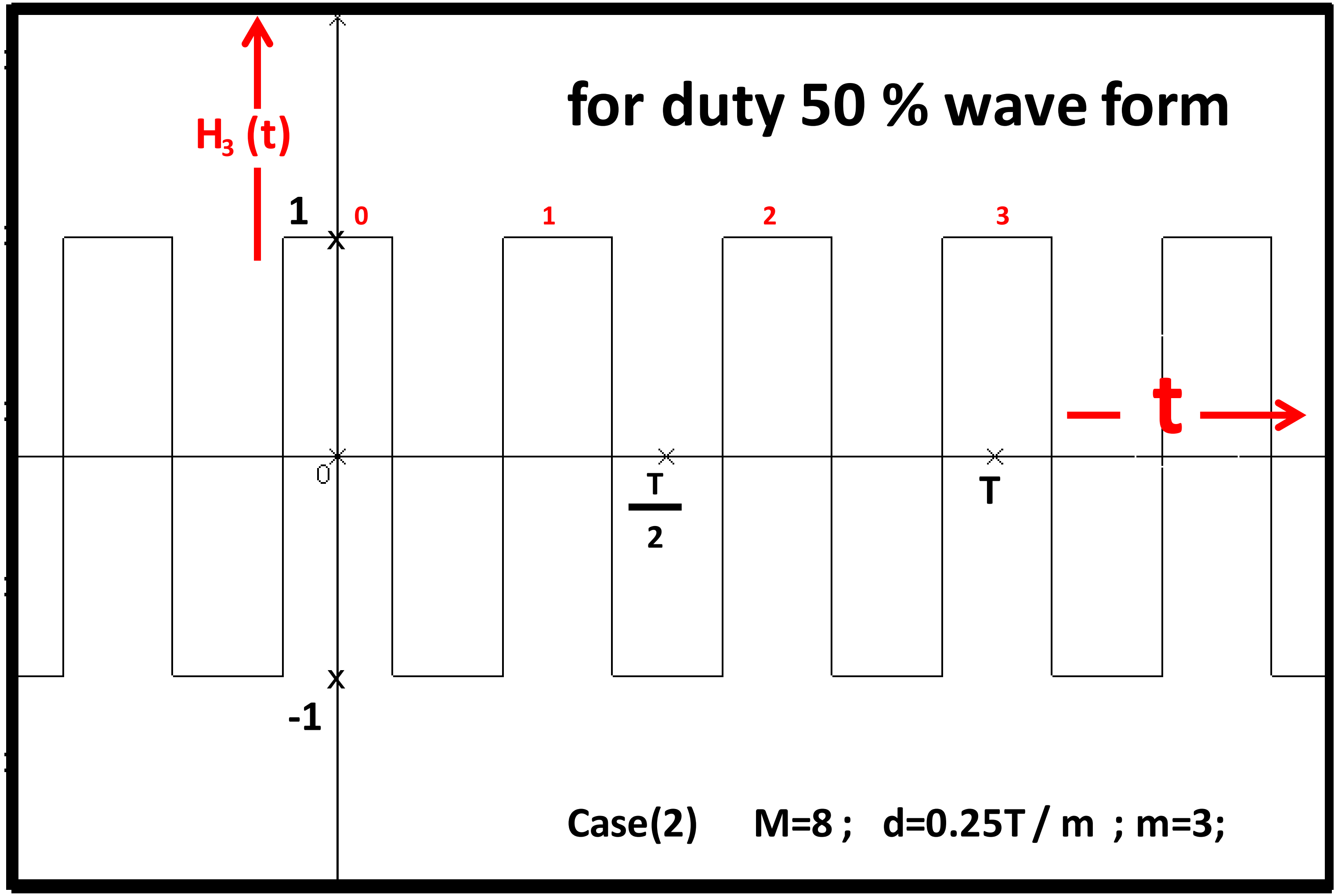
Make the pulse very narrow.

Case(1) M=8 ; d=0.05T / M ; m=2;

Nonuniform T/k Sampling Scheme
with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 3;

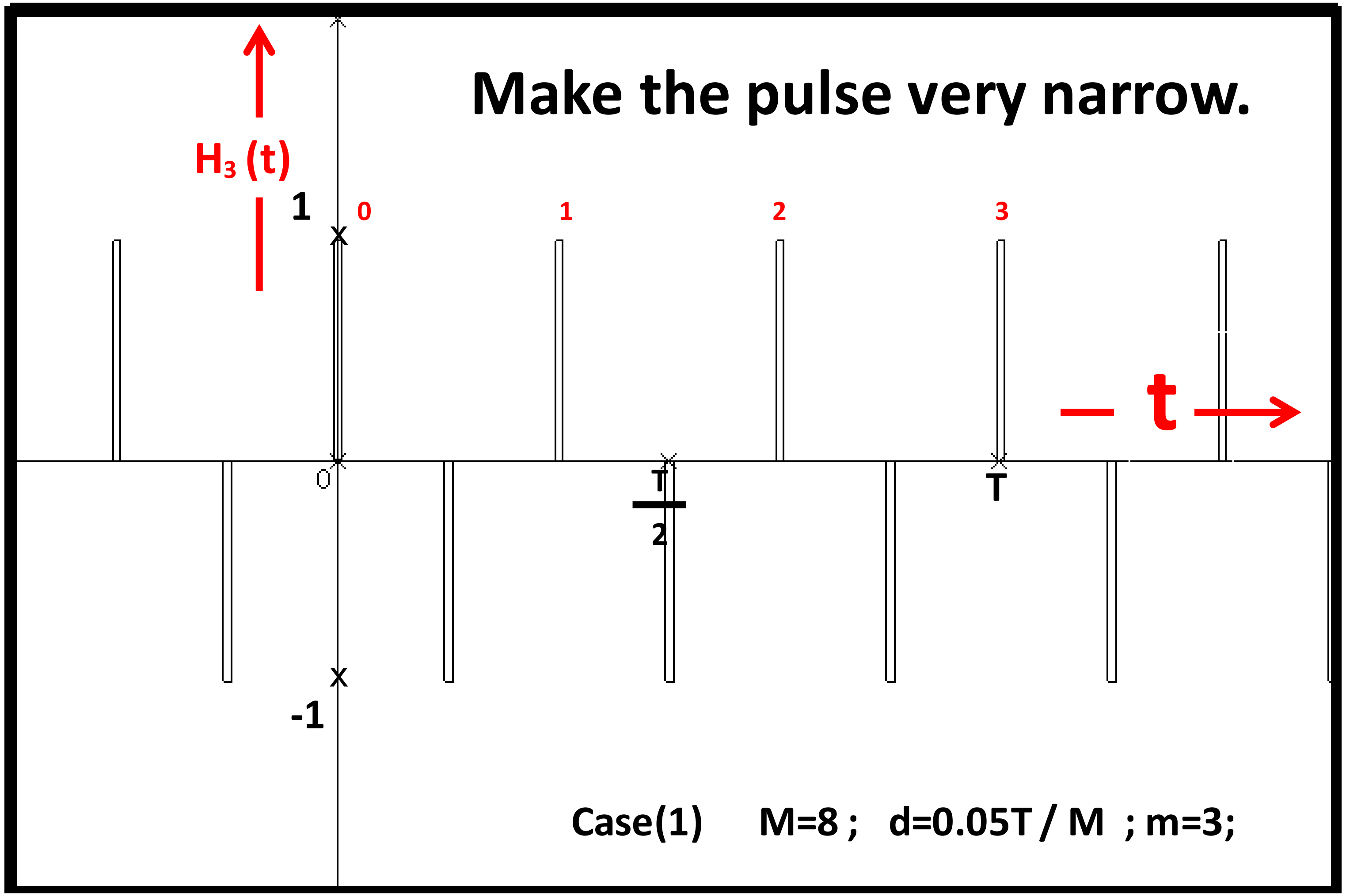
for N = 8 points



Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 3;

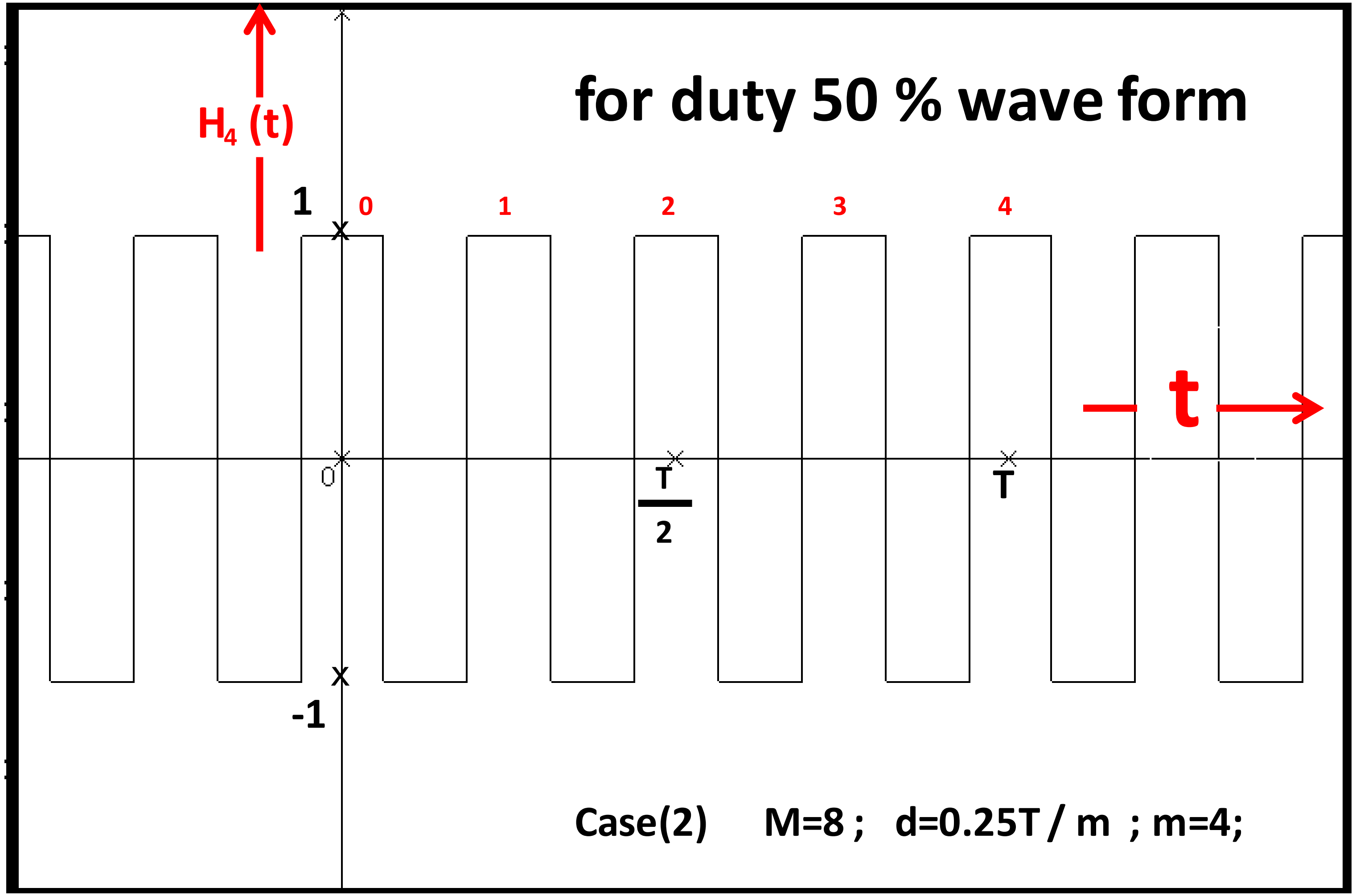
for N = 8 points



Nonuniform T/k Sampling Scheme
with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 4;

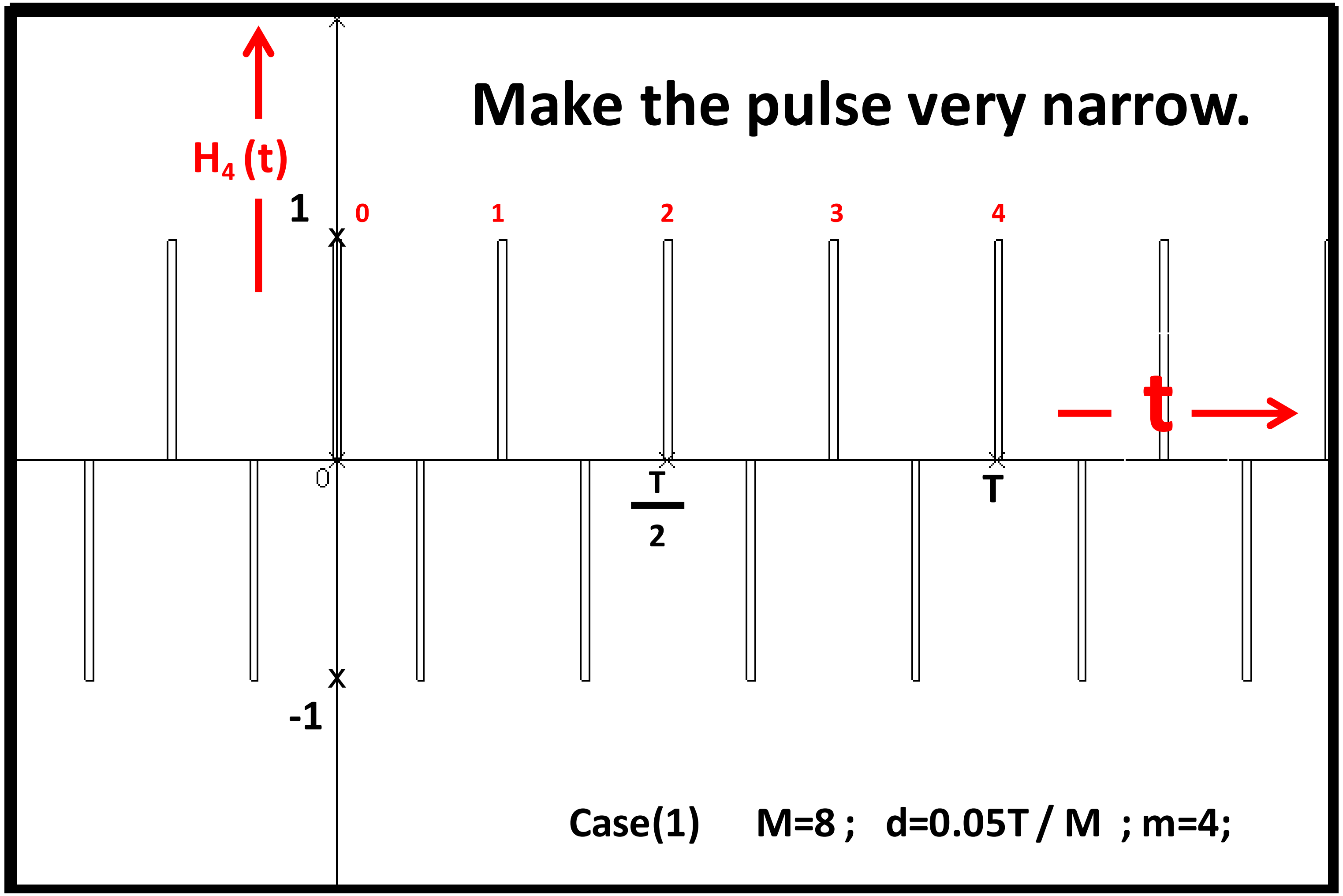
for N = 8 points



Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 4;

for N = 8 points



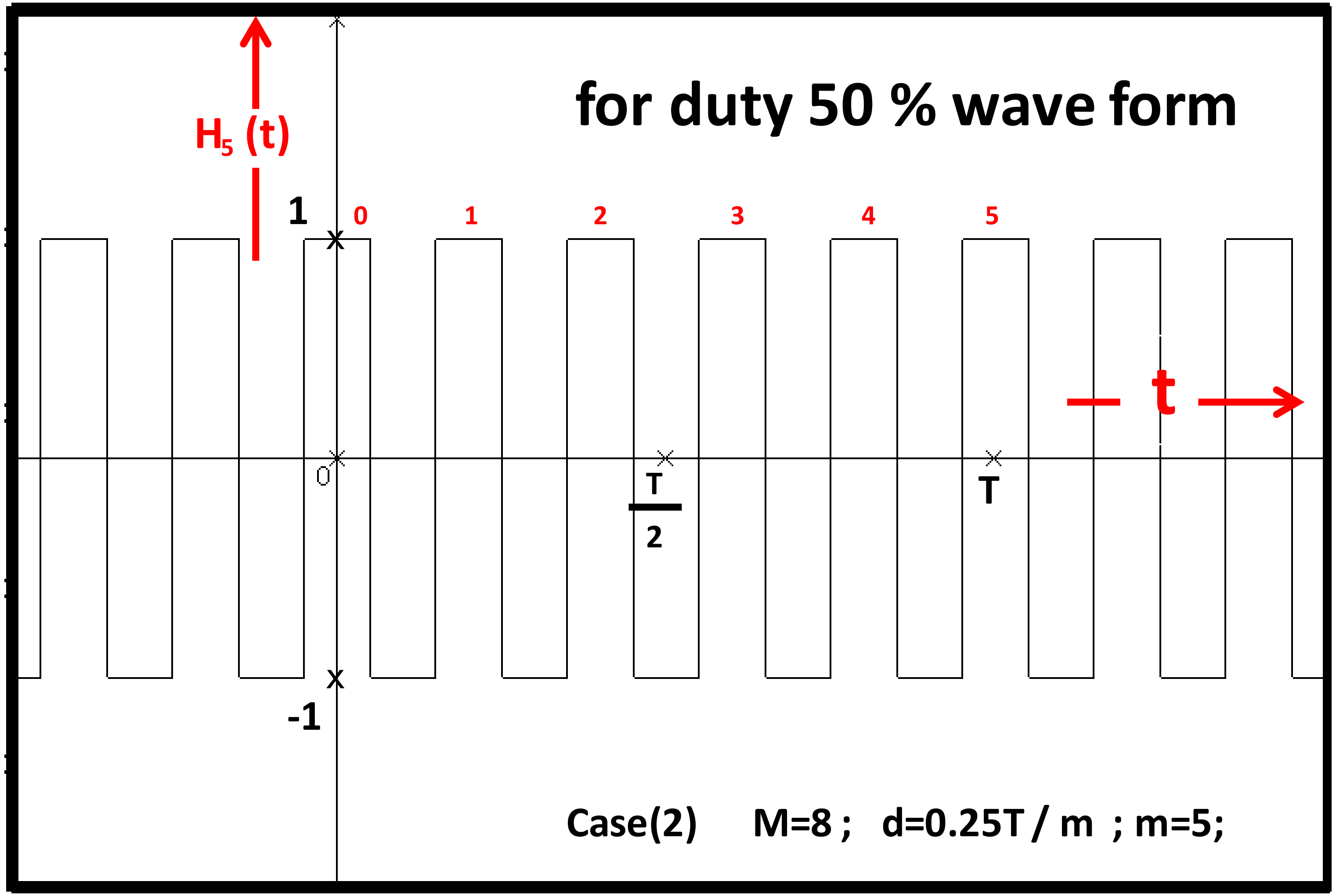
Make the pulse very narrow.

Case(1) M=8 ; d=0.05T / M ; m=4;

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 5;

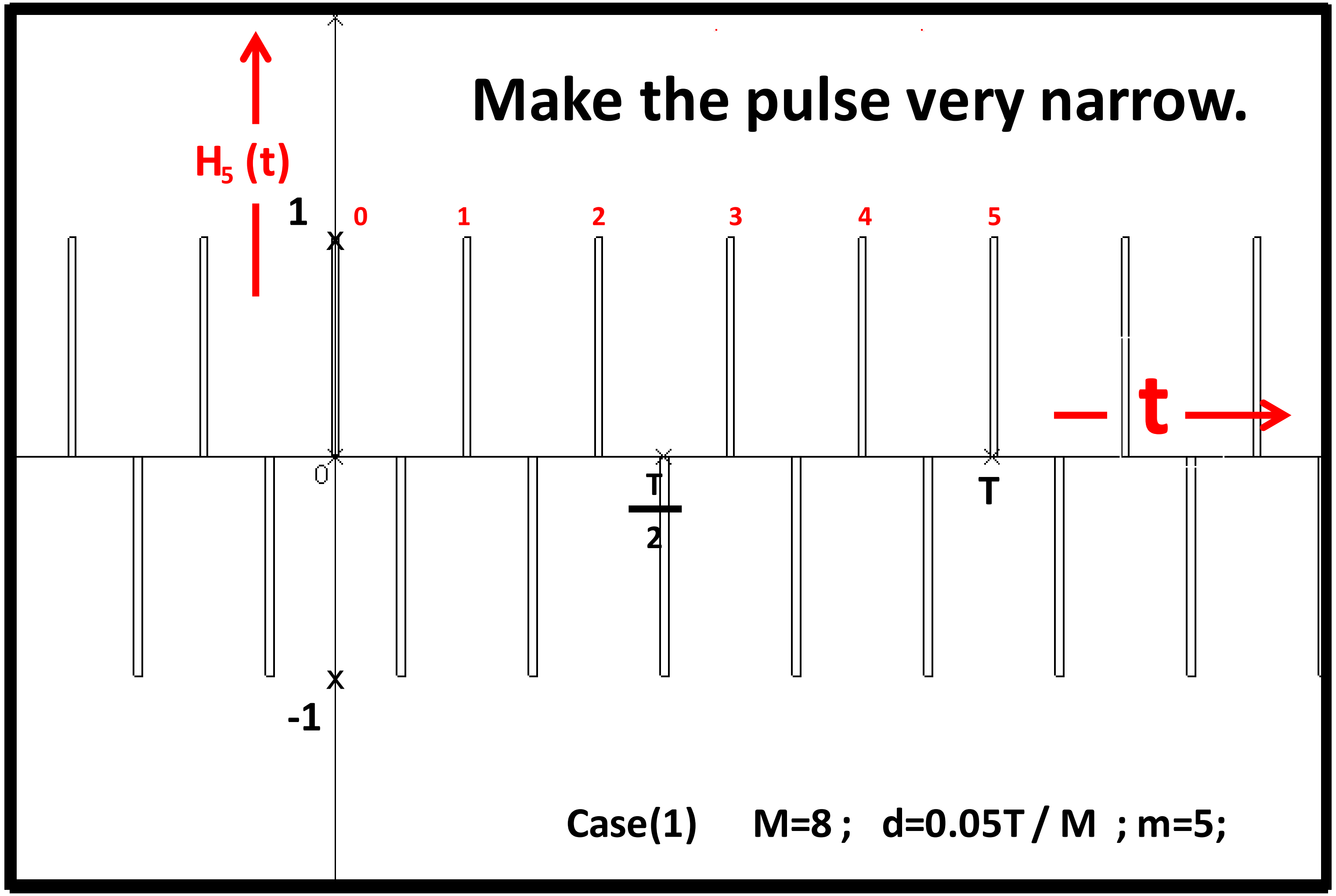
for N = 8 points



Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 5;

for N = 8 points



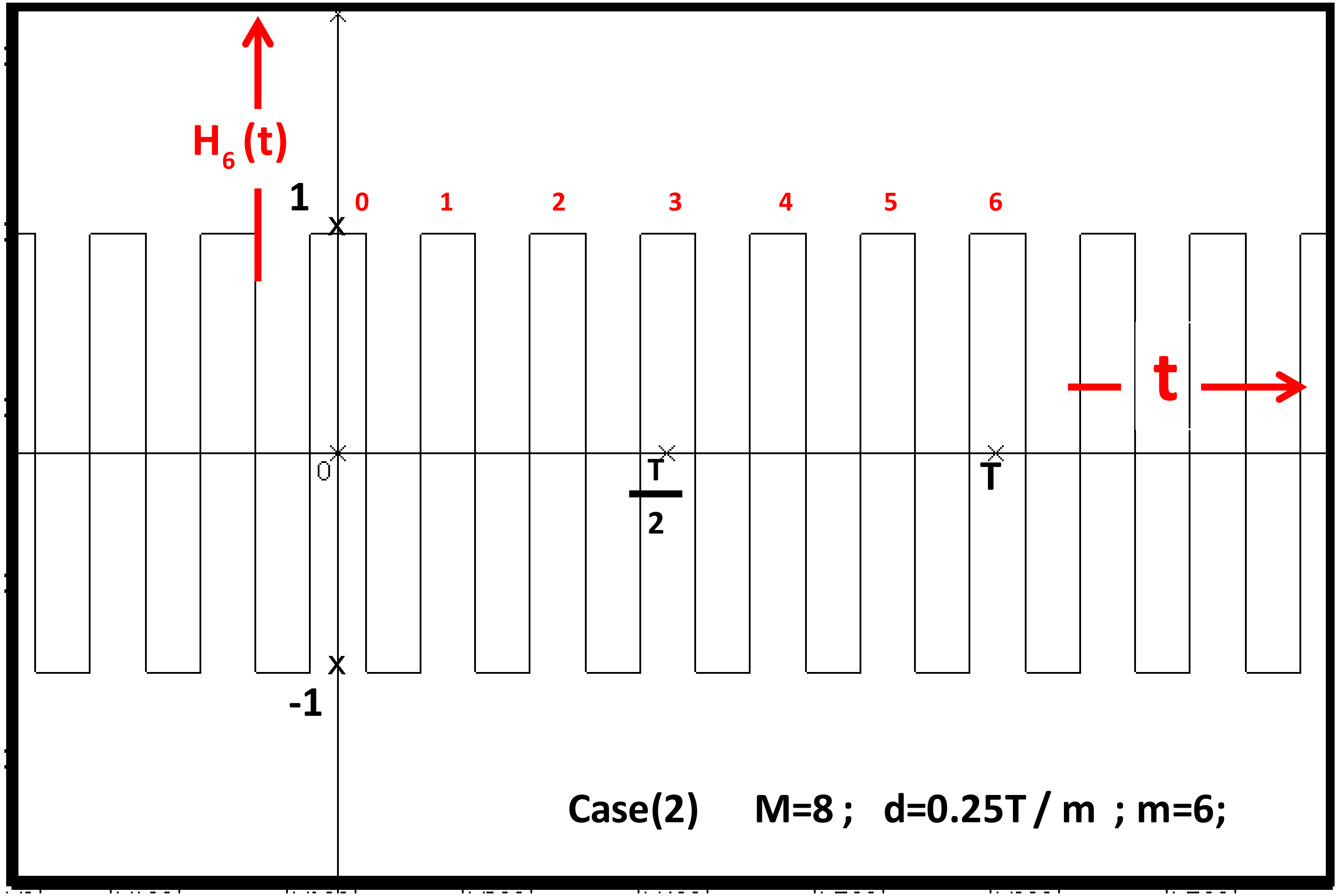
Make the pulse very narrow.

Case(1) M=8 ; d=0.05T / M ; m=5;

Nonuniform T/k Sampling Scheme
with Fast H [ ][ ] Transformation Matrix

$H_m(t)$ function; m = 6;

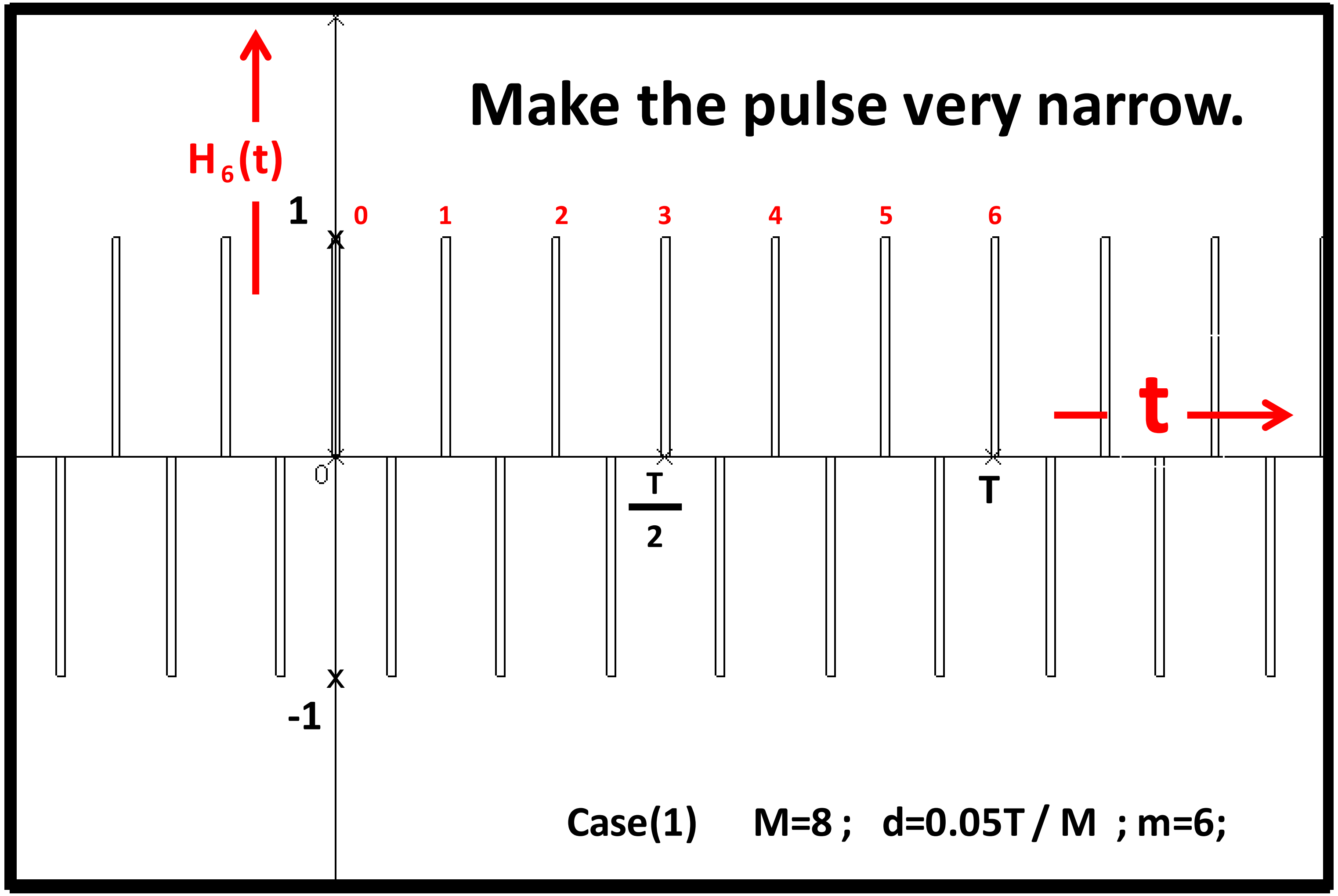
for N = 8 points



Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 6;

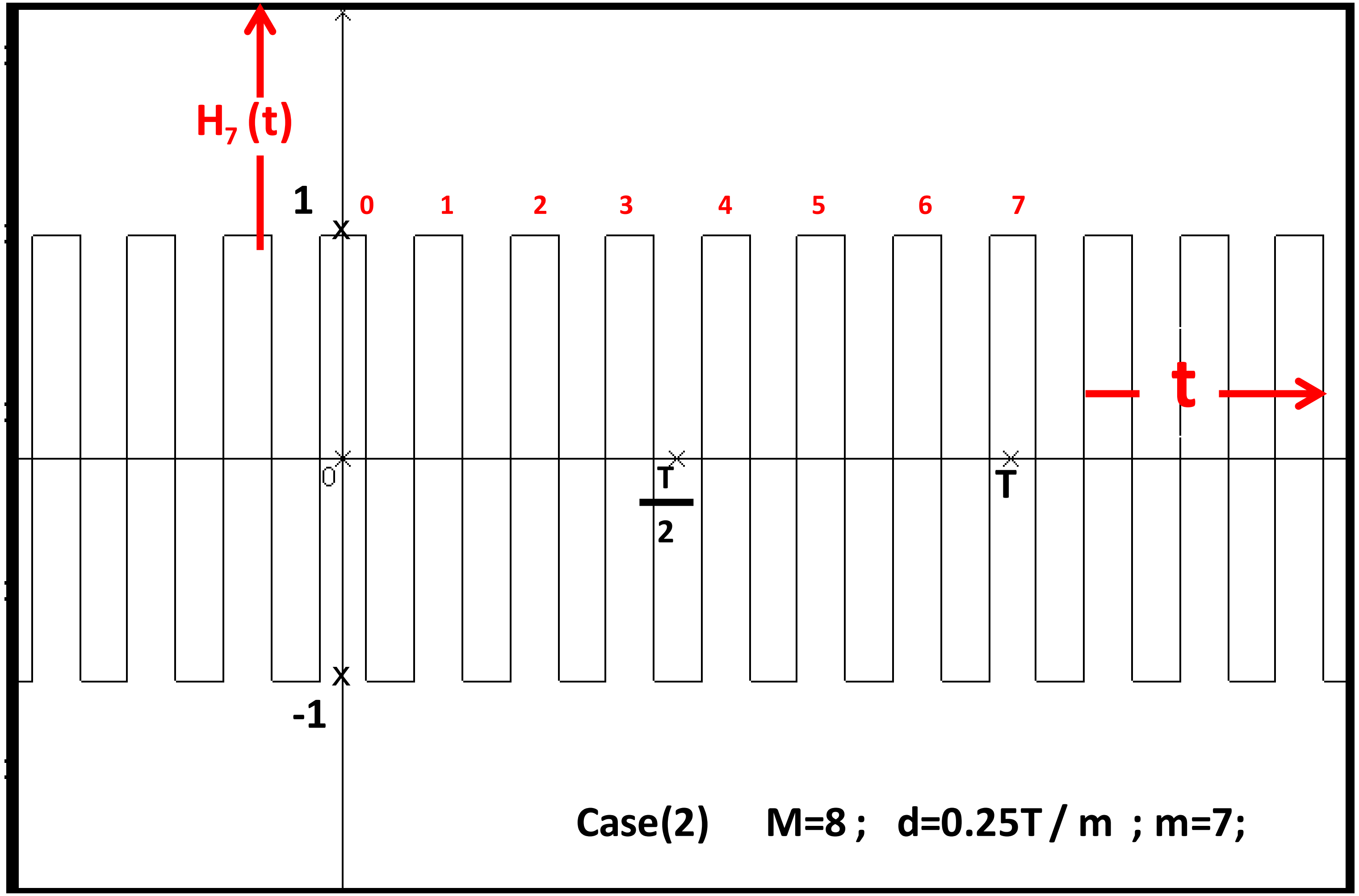
for N = 8 points



Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

H_m (t) function; m = 7;

for N = 8 points

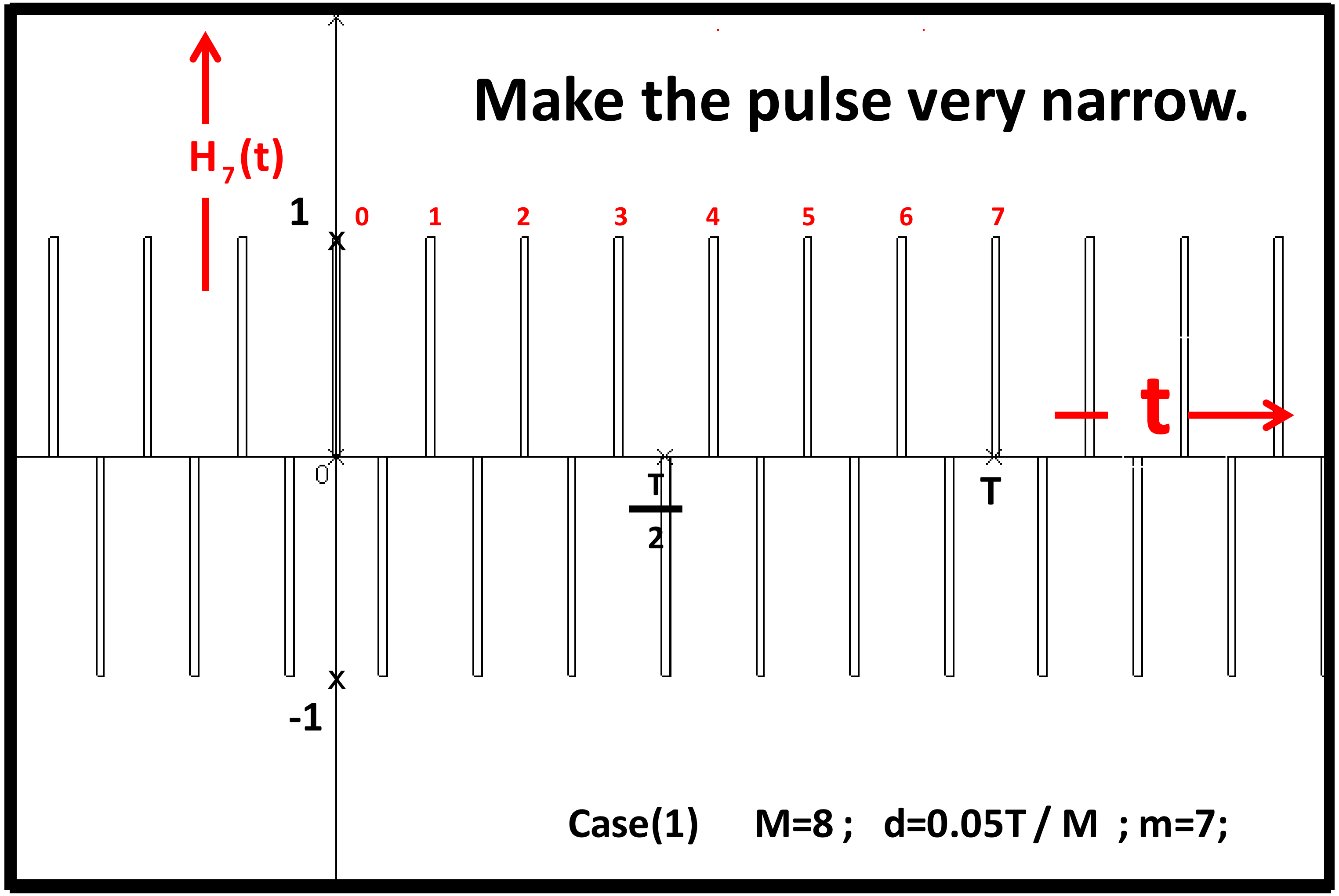


Case(2) M=8 ; d=0.25T / m ; m=7;

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 7;

for N = 8 points

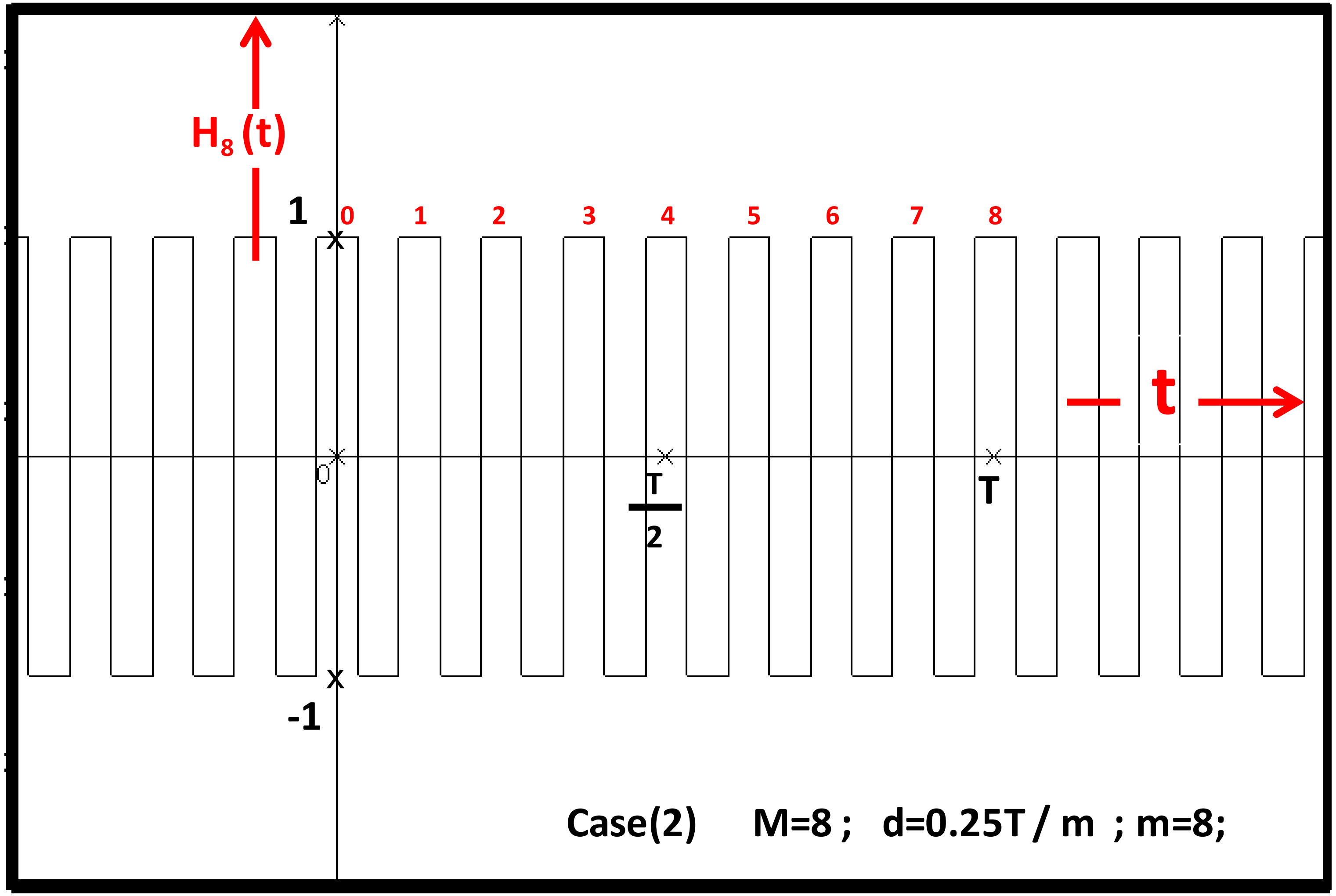


Case(1) M=8 ; d=0.05T / M ; m=7;

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

$H_m(t)$ function ; m = 8;

for N = 8 points

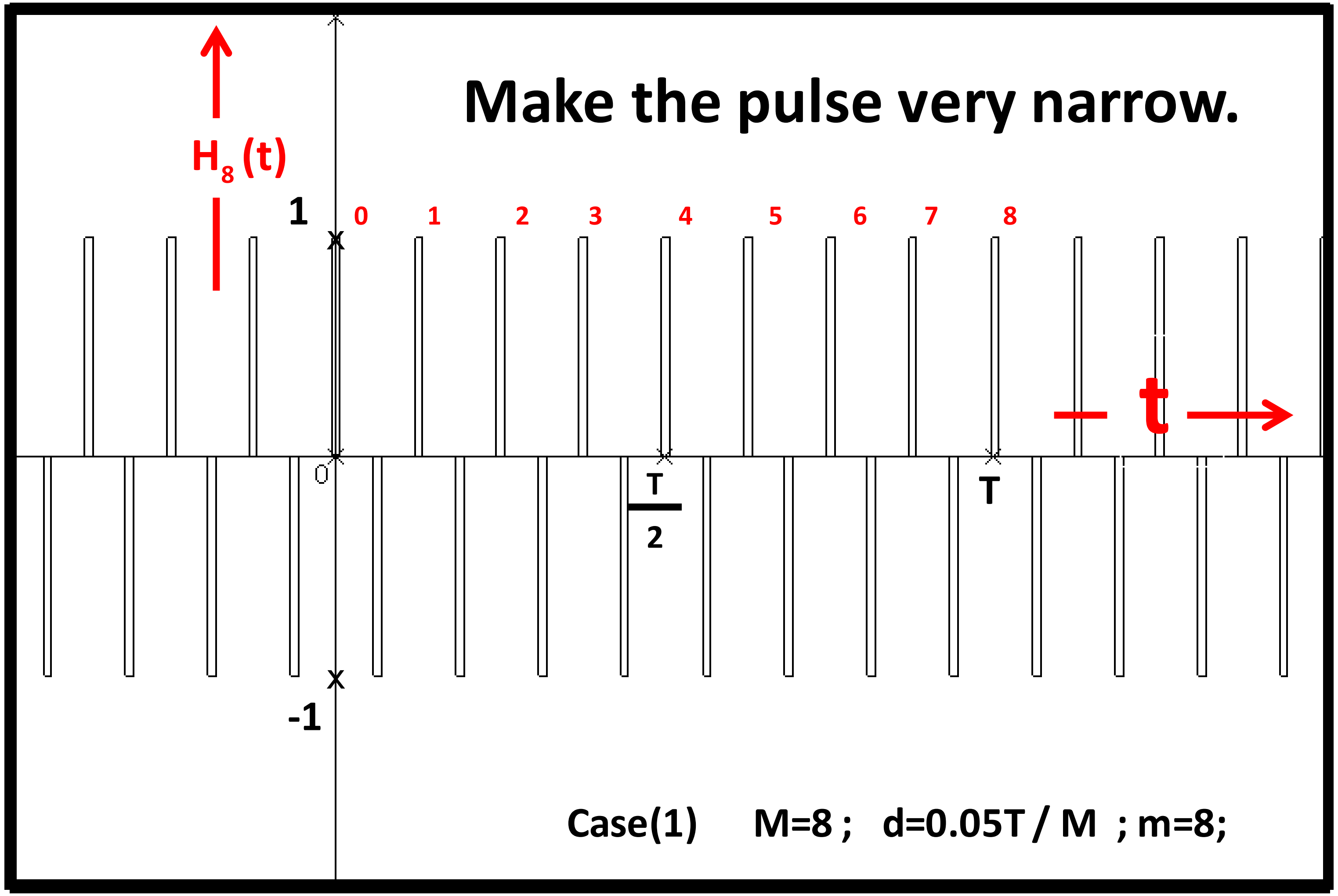


Case(2) M=8 ; d=0.25T / m ; m=8;

Nonuniform T/k Sampling Scheme
with Fast H [ ][ ] Transformation Matrix

Define $H_m(t)$ function ; m = 8;

for N = 8 points



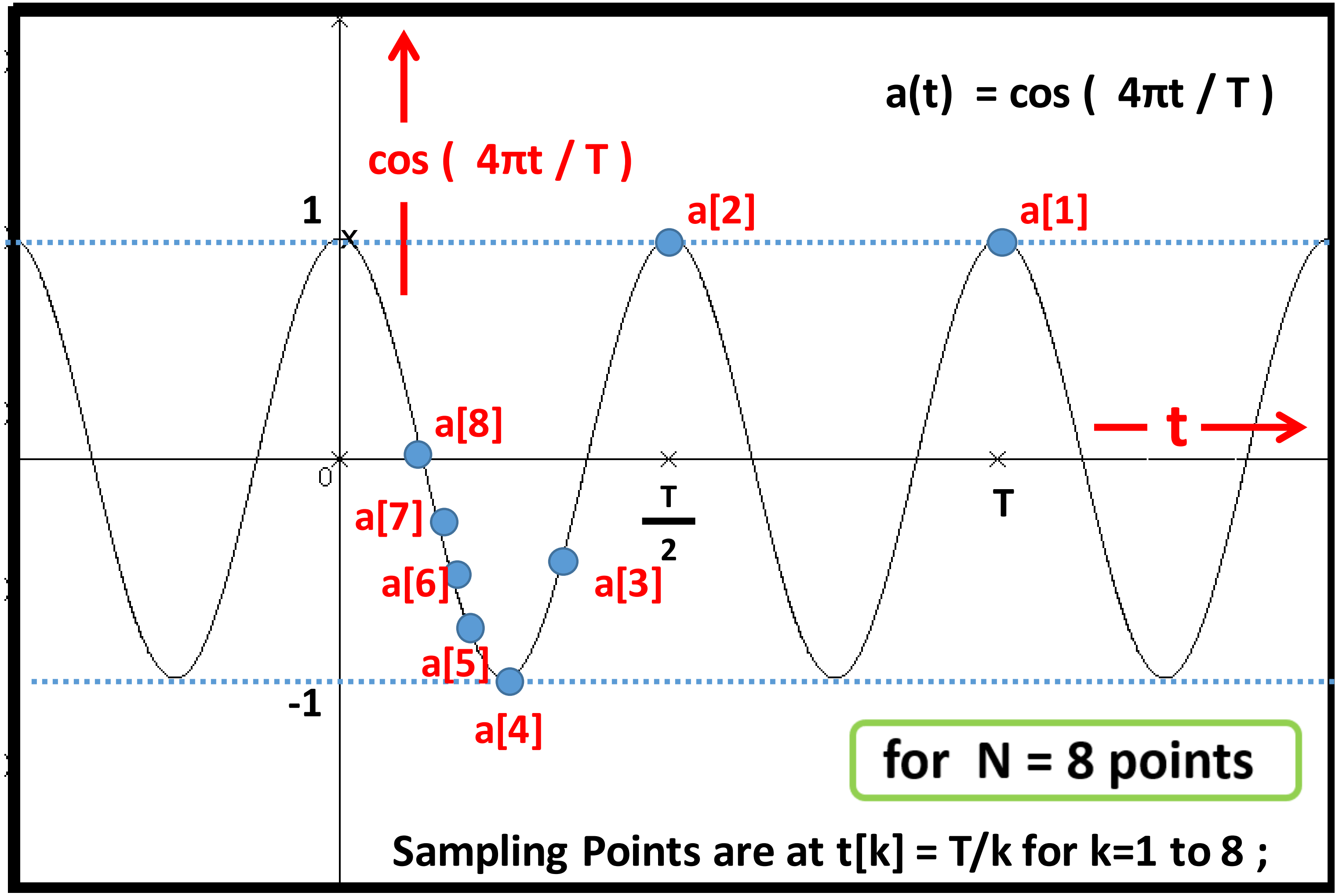
Case(1) M=8 ; d=0.05T / M ; m=8;

Step(1) Define $H_m(t)$ functions
Step(2) Define H[m][k]

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

a[1]= 1
a[2]= 1
a[3]= - 0.5
a[4]= - 1
a[5]= - 0.809
a[6]= - 0.5
a[7]= - 0.223
a[8]= 0

A[1]= 1.032
A[2]= 1.5
A[3]= 0
A[4]= 0
A[5]= - 0.809
A[6]= - 0.5
A[7]= - 0.223
A[8]= 0



for N = 8 points

Sampling Points are at t[k] = T/k for k=1 to 8 ;

Compute $A[m] = \sum_{k=1}^{8} (1/8)\, H[m][k]\, a[k]$; Then we have $a(t) = \sum_{m=1}^{8} A[m]\, H_k(t)$;

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

We only need the bit shifter circuits for the matrix calculations.

$$a[\ ] = H[\ ][\ ]A[\ ]; \quad \text{for } N = 1;$$

$$a[\,1\,] = [\,1\,]\,A[\,1\,]$$

$$A[\ ] = H^{-1}[\ ][\ ]\,a[\ ];$$

$$A[\,1\,] = (1/1)\,[1]\,a[\,1\,]$$

Nonuniform T/k Sampling Scheme
with Fast H [ ][ ] Transformation Matrix

We only need the bit shifter circuits for the matrix calculations.

a[ ] = H [ ] [ ]A[ ] ; for N =2 ;

$$\begin{vmatrix} a[1] \\ a[2] \end{vmatrix} = \begin{vmatrix} 1 & 1 \\ -1 & 1 \end{vmatrix} \begin{vmatrix} A[1] \\ A[2] \end{vmatrix}$$

$A[\ ] = H^{-1}[\ ][\ ]\, a[\ ]$;

$$\begin{vmatrix} A[1] \\ A[2] \end{vmatrix} = (1/2) \begin{vmatrix} 1 & -1 \\ 1 & 1 \end{vmatrix} \begin{vmatrix} a[1] \\ a[2] \end{vmatrix}$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

We only need the bit shifter circuits for the matrix calculations.

$$a[\ ] = H[\ ][\ ]A[\ ]; \quad \text{for } N = 3;$$

$$\begin{vmatrix} a[1] \\ a[2] \\ a[3] \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ -1 & 1 & -1 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} A[1] \\ A[2] \\ A[3] \end{vmatrix}$$

$$A[\ ] = H^{-1}[\ ][\ ]\,a[\ ];$$

$$\begin{vmatrix} A[1] \\ A[2] \\ A[3] \end{vmatrix} = (1/2) \begin{vmatrix} 1 & -1 & -2 \\ 1 & 1 & 0 \\ 0 & 0 & 2 \end{vmatrix} \begin{vmatrix} a[1] \\ a[2] \\ a[3] \end{vmatrix}$$

Nonuniform T/k Sampling Scheme
with Fast H [ ][ ] Transformation Matrix

We only need the bit shifter circuits for the matrix calculations.

$$a[\ ] = H[\ ][\ ]A[\ ]; \quad \text{for } N = 4 ;$$

$$\begin{vmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{vmatrix} \begin{vmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \end{vmatrix}$$

$$A[\ ] = H^{-1}[\ ][\ ]\, a[\ ] ;$$

$$\begin{vmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \end{vmatrix} = (1/4) \begin{vmatrix} 2 & -2 & -4 & 0 \\ 1 & 1 & 0 & -2 \\ 0 & 0 & 4 & 0 \\ 1 & 1 & 0 & 2 \end{vmatrix} \begin{vmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \end{vmatrix}$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

We only need the bit shifter circuits for the matrix calculations.

a[ ] = H [ ] [ ]A[ ] ; for N =5 ;

$$\begin{vmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & -1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \\ A[5] \end{vmatrix}$$

A[ ] = H^{-1}[ ][ ] a[ ] ;

$$\begin{vmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \\ A[5] \end{vmatrix} = (1/4) \begin{vmatrix} 2 & -2 & -4 & 0 & -4 \\ 1 & 1 & 0 & -2 & 0 \\ 0 & 0 & 4 & 0 & 0 \\ 1 & 1 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 4 \end{vmatrix} \begin{vmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \end{vmatrix}$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

We only need the bit shifter circuits for the matrix calculations.

$$a[\ ] = H[\ ][\ ]A[\ ]; \quad \text{for } N = 6;$$

$$\left|\begin{matrix} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \\ a[6] \end{matrix}\right| = \left|\begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & -1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{matrix}\right| \left|\begin{matrix} A[1] \\ A[2] \\ A[3] \\ A[4] \\ A[5] \\ A[6] \end{matrix}\right|$$

$$A[\ ] = H^{-1}[\ ][\ ]\,a[\ ];$$

$$\left|\begin{matrix} A[1] \\ A[2] \\ A[3] \\ A[4] \\ A[5] \\ A[6] \end{matrix}\right| = (1/4)\left|\begin{matrix} 2 & -2 & -2 & 0 & -4 & 2 \\ 1 & 1 & -2 & -2 & 0 & -2 \\ 0 & 0 & 2 & 0 & 0 & -2 \\ 1 & 1 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 2 & 0 & 0 & 2 \end{matrix}\right| \left|\begin{matrix} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \\ a[6] \end{matrix}\right|$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

We only need the bit shifter circuits for the matrix calculations.

$$a[\ ] = H[\ ][\ ]A[\ ]; \quad \text{for } N = 7;$$

$$\left|\begin{array}{c} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \\ a[6] \\ a[7] \end{array}\right| = \left|\begin{array}{ccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right| \left|\begin{array}{c} A[1] \\ A[2] \\ A[3] \\ A[4] \\ A[5] \\ A[6] \\ A[7] \end{array}\right|$$

$$A[\ ] = H^{-1}[\ ][\ ]\, a[\ ];$$

$$\left|\begin{array}{c} A[1] \\ A[2] \\ A[3] \\ A[4] \\ A[5] \\ A[6] \\ A[7] \end{array}\right| = (1/4) \left|\begin{array}{ccccccc} 2 & -2 & -2 & 0 & -4 & 2 & -4 \\ 1 & 1 & -2 & -2 & 0 & -2 & 0 \\ 0 & 0 & 2 & 0 & 0 & -2 & 0 \\ 1 & 1 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 \end{array}\right| \left|\begin{array}{c} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \\ a[6] \\ a[7] \end{array}\right|$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

We only need the bit shifter circuits for the matrix calculations.

$$a[\ ] = H[\ ][\ ]A[\ ]; \quad \text{for } N = 8;$$

$$\begin{vmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \\ a[6] \\ a[7] \\ a[8] \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} A[1] \\ A[2] \\ A[3] \\ A[4] \\ A[5] \\ A[6] \\ A[7] \\ A[8] \end{vmatrix}$$

$$A[\ ] = H^{-1}[\ ][\ ]\, a[\ ];$$

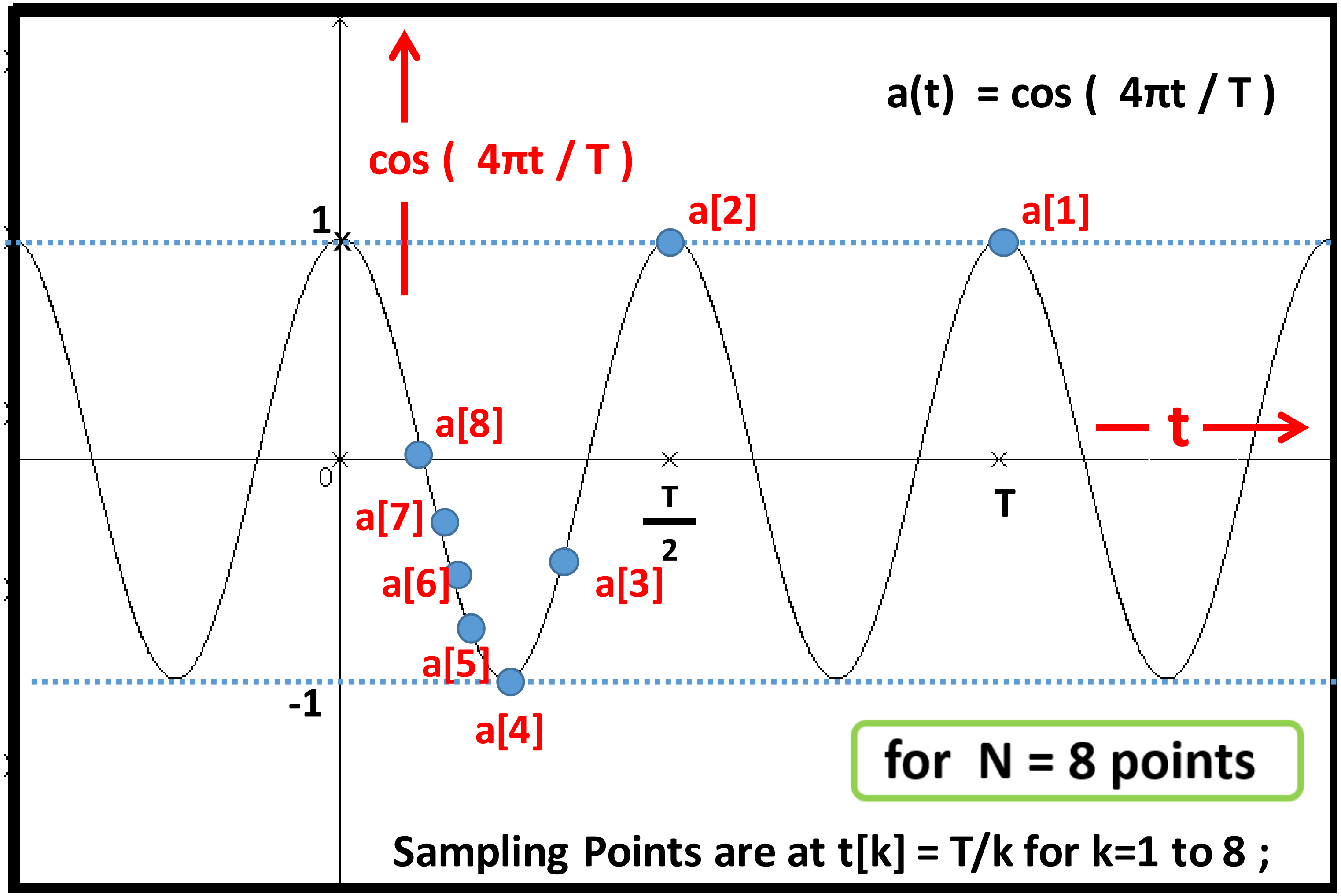
$$\begin{vmatrix} A[1] \\ A[2] \\ A[3] \\ A\,4] \\ A[5] \\ A[6] \\ A[7] \\ A[8] \end{vmatrix} = (1/8) \begin{vmatrix} 4 & -4 & -4 & 0 & -8 & 4 & -8 & 0 \\ 2 & 2 & -4 & -4 & 0 & -4 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & -4 & 0 & 0 \\ 1 & 1 & 0 & 2 & 0 & 0 & 0 & -4 \\ 0 & 0 & 0 & 0 & 8 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 8 & 0 \\ 1 & 1 & 0 & 2 & 0 & 0 & 0 & 4 \end{vmatrix} \begin{vmatrix} a[1] \\ a[2] \\ a[3] \\ a[4] \\ a[5] \\ a[6] \\ a[7] \\ a[8] \end{vmatrix}$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Step(1) Define $H_m(t)$ functions
Step(2) Define H[m][k]
Step(3) Obtain a[m]
Step(4) Obtain A[m]

a[1]= 1
a[2]= 1
a[3]= - 0.5
a[4]= - 1
a[5]= - 0.809
a[6]= - 0.5
a[7]= - 0.223
a[8]= 0

A[1]= 1.032
A[2]= 1.5
A[3]= 0
A[4]= 0
A[5]= - 0.809
A[6]= - 0.5
A[7]= - 0.223
A[8]= 0



a(t) = cos (4πt / T)

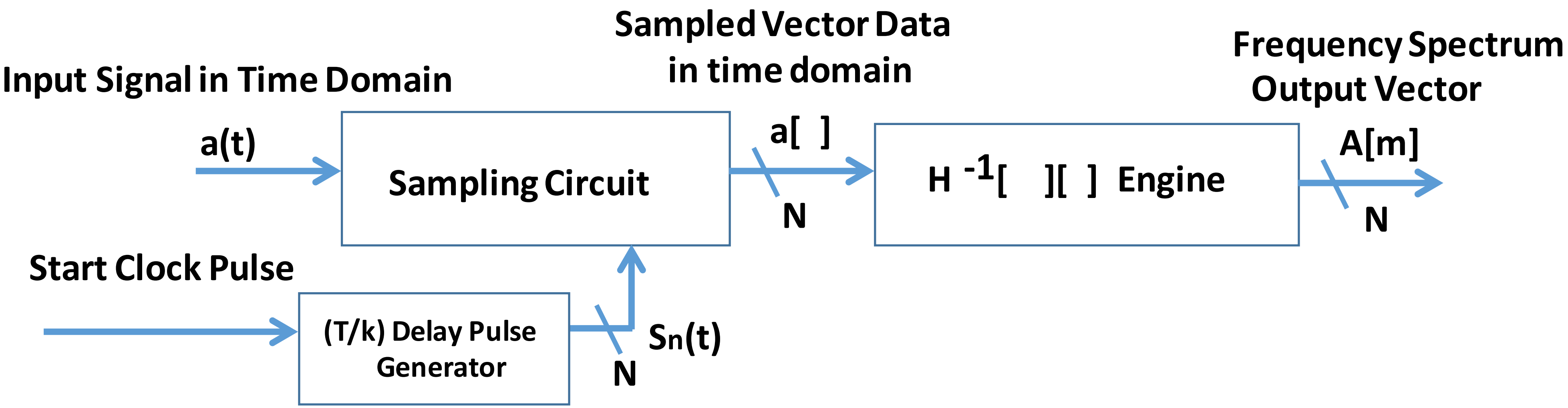
for N = 8 points

Sampling Points are at t[k] = T/k for k=1 to 8 ;

Compute $A[m] = \sum_{k=1}^{8} (1/8)\, H[m][k]\, a[k]$; Then we have $a(t) = \sum_{m=1}^{8} A[m]\, H_m(t)$;

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Block Diagram of (T/k) Delay Pulse Generator



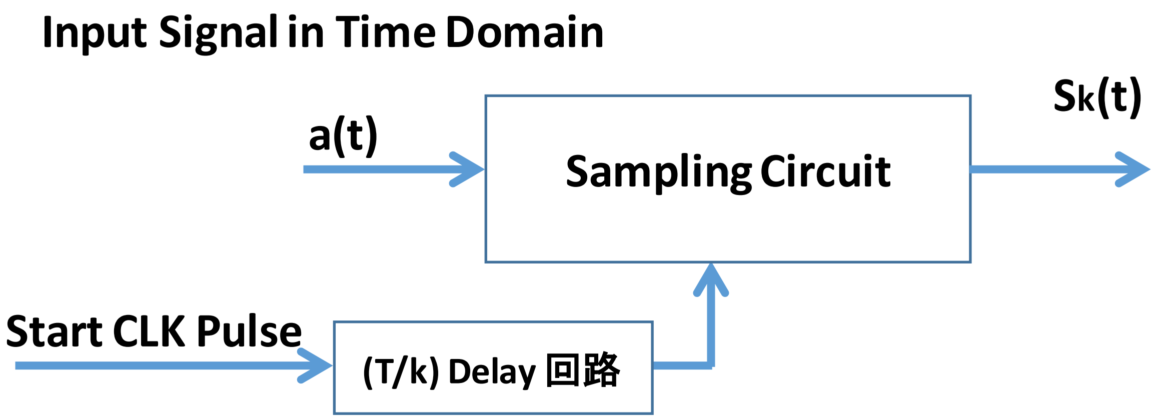
Input Signal in Time Domain
a(t)
Sampling Circuit
Sampled Vector Data in time domain
a[ ]
N
H -1[ ][ ] Engine
Frequency Spectrum Output Vector
A[m]
N
Start Clock Pulse
(T/k) Delay Pulse Generator
N
Sn(t)

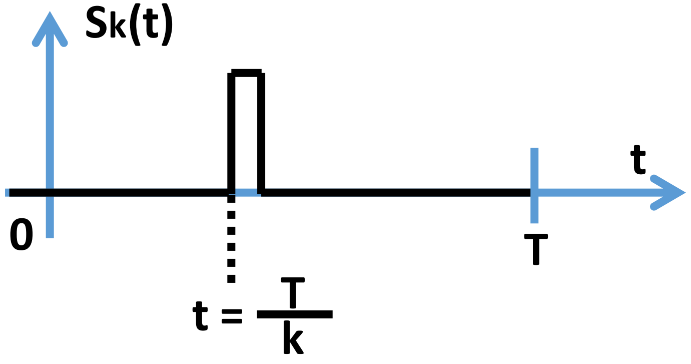
$$a[\ ]\ =\ H[\ ][\ ]A[\ ];$$

$$A[\ ] = H^{-1}[\ ][\ ]\,a[\ ]\ ;$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Block Diagram of (T/k) Delay Pulse Generator



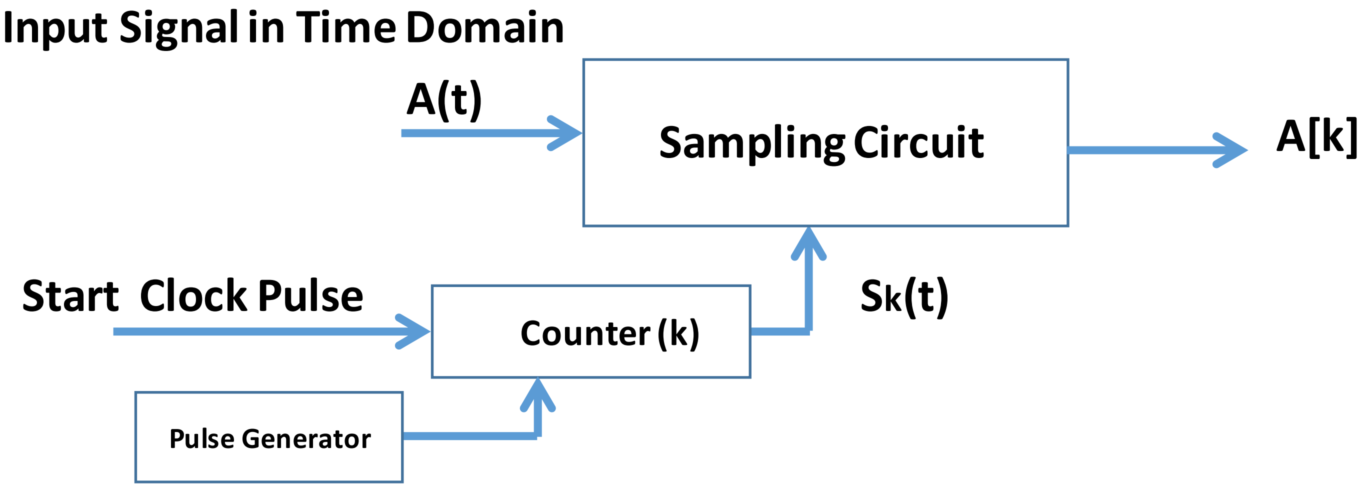


$$a[\ ] = H[\ ][\ ]A[\ ];$$

$$A[\ ] = H^{-1}[\ ][\ ]\,a[\ ];$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Block Diagram of (T/k) Delay Pulse Generator

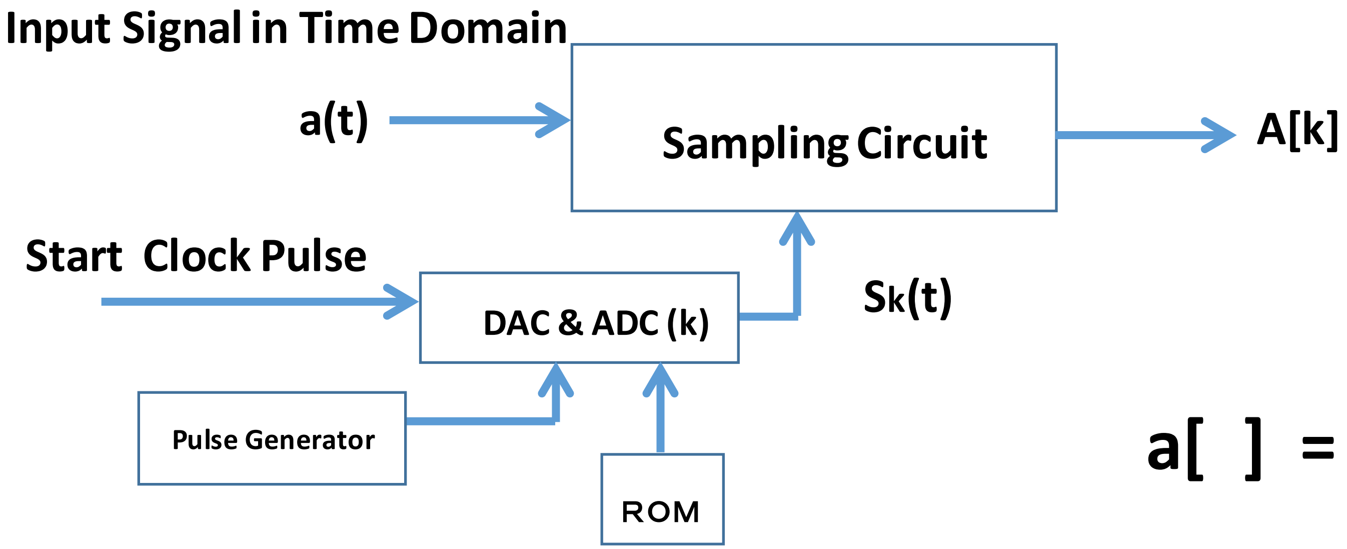


$$a[\ ] = H[\ ][\ ]A[\ ];$$

$$A[\ ] = H^{-1}[\ ][\ ]\,a[\ ];$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Block Diagram of (T/k) Delay Pulse Generator

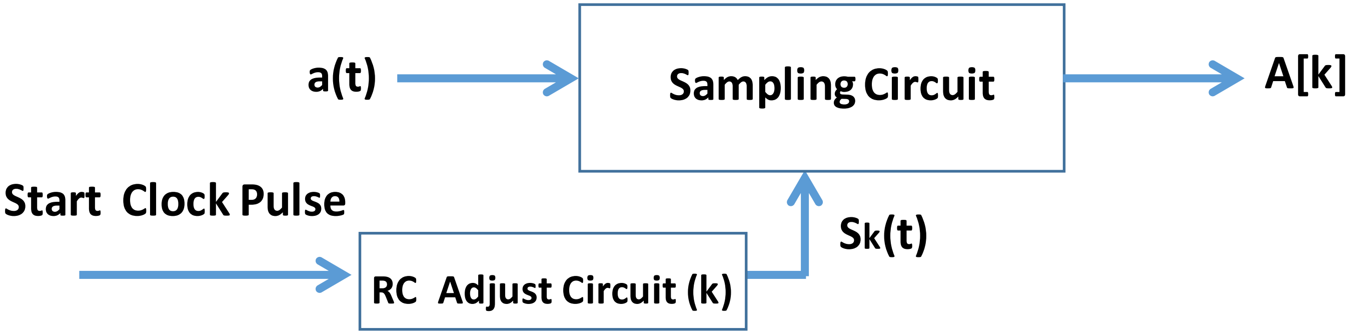


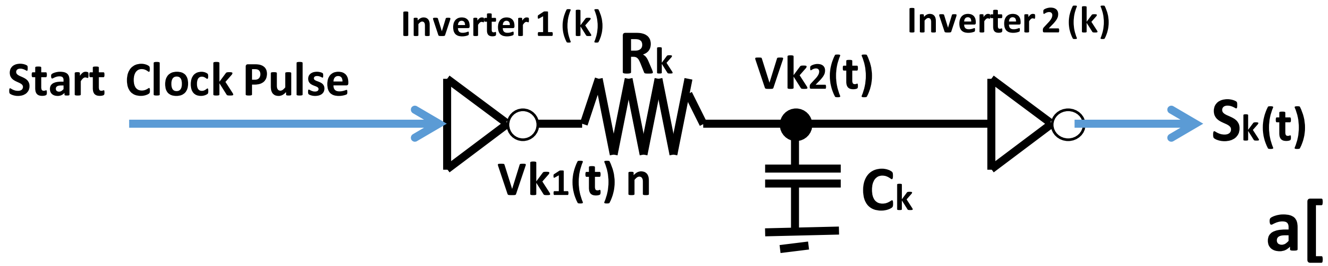
$$a[\ ] = H[\ ][\ ]A[\ ];$$

$$A[\ ] = H^{-1}[\ ][\ ]\,a[\ ]\ ;$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Block Diagram of (T/k) Delay Pulse Generator





$$a[\ ] = H[\ ][\ ]A[\ ];$$

$$A[\ ] = H^{-1}[\ ][\ ]\,a[\ ]\ ;$$

Nonuniform T/k Sampling Scheme with Fast H [ ][ ] Transformation Matrix

Step(1) Define $H_m(t)$ functions

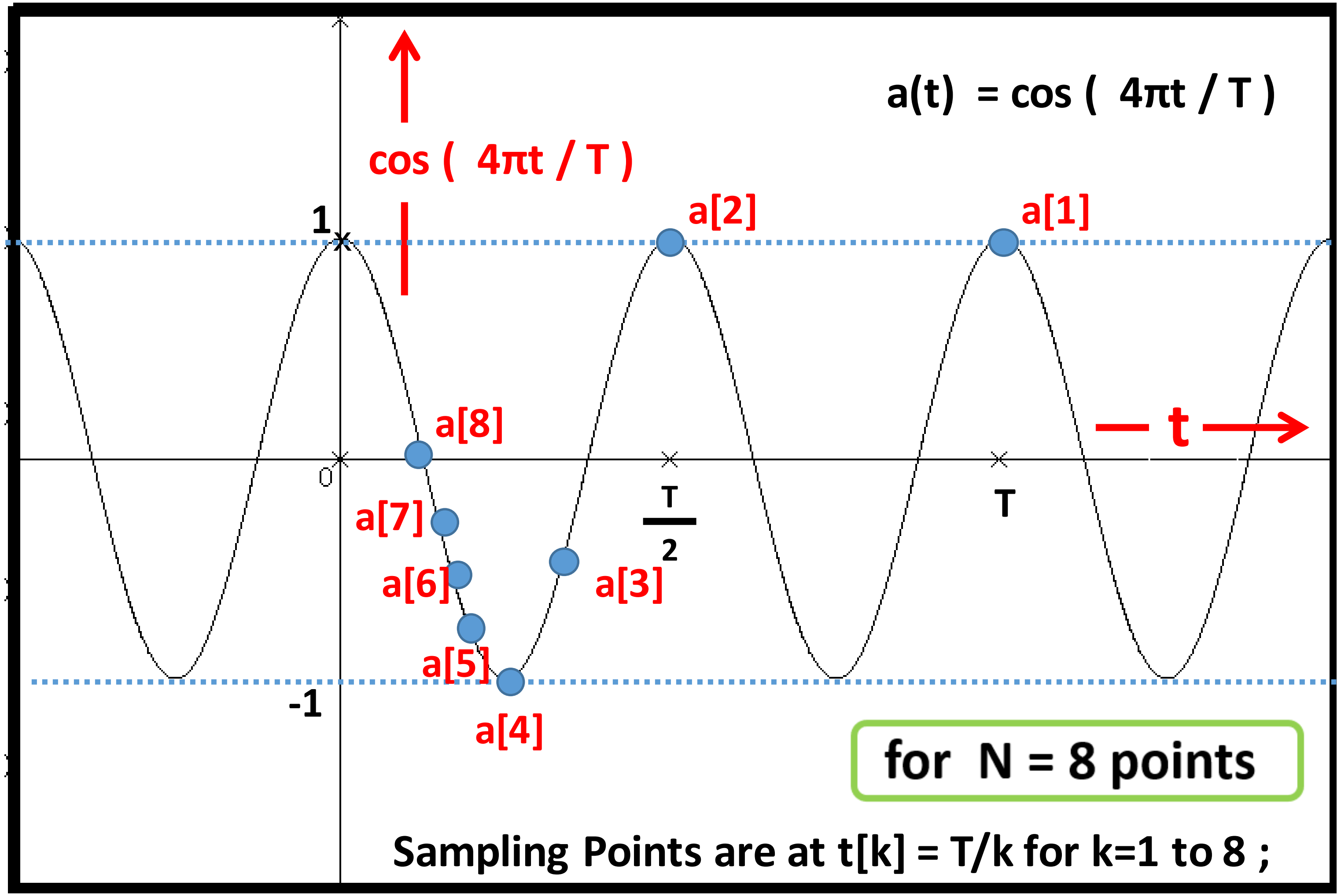
Step(2) Define H[m][k]

Step(3) Obtain a[m]

Step(4) Obtain A[m]

a[1]= 1
a[2]= 1
a[3]= - 0.5
a[4]= - 1
a[5]= - 0.809
a[6]= - 0.5
a[7]= - 0.223
a[8]= 0

A[1]= 1.032
A[2]= 1.5
A[3]= 0
A[4]= 0
A[5]= - 0.809
A[6]= - 0.5
A[7]= - 0.223
A[8]= 0

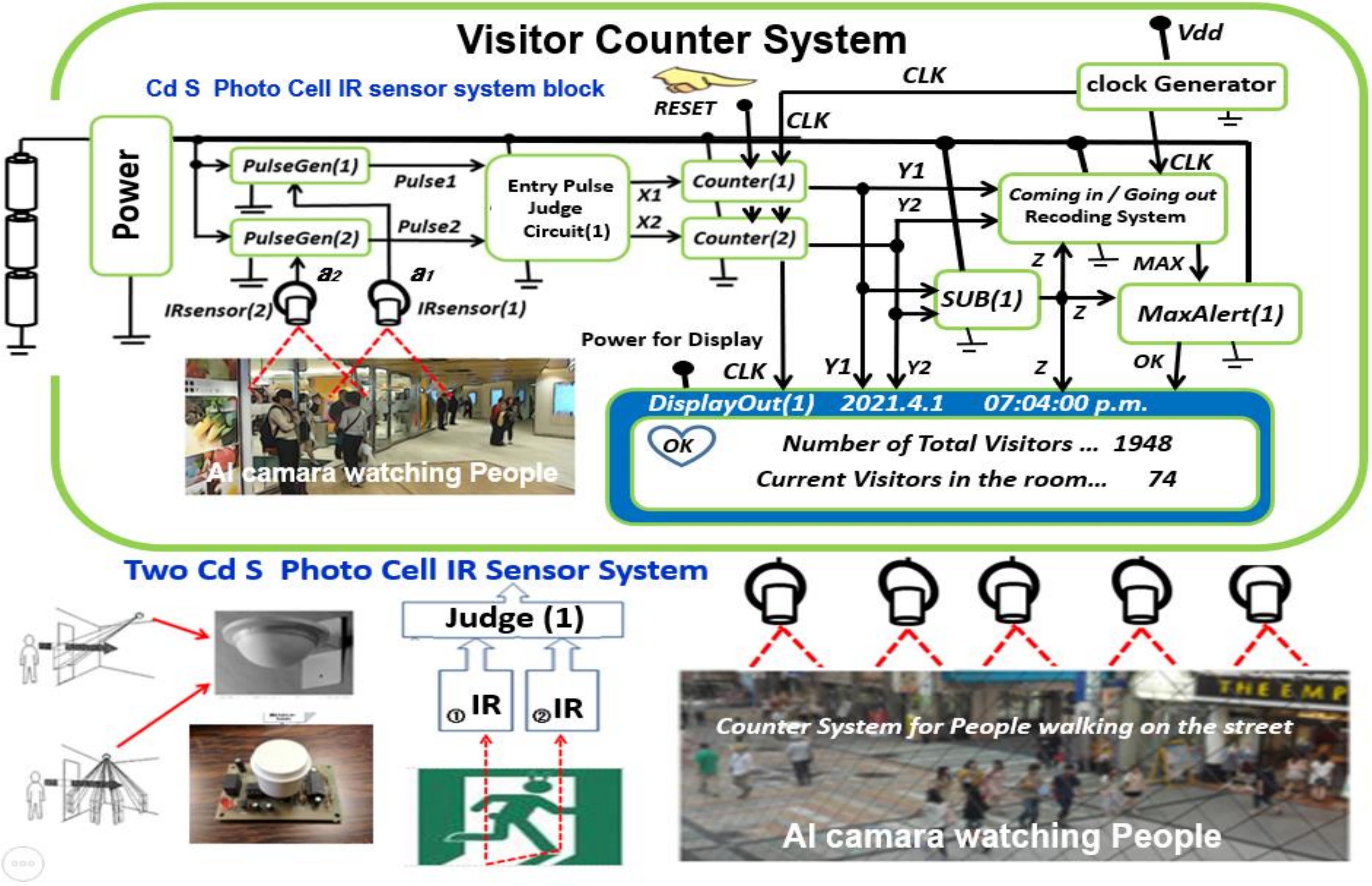


for N = 8 points

Sampling Points are at t[k] = T/k for k=1 to 8 ;

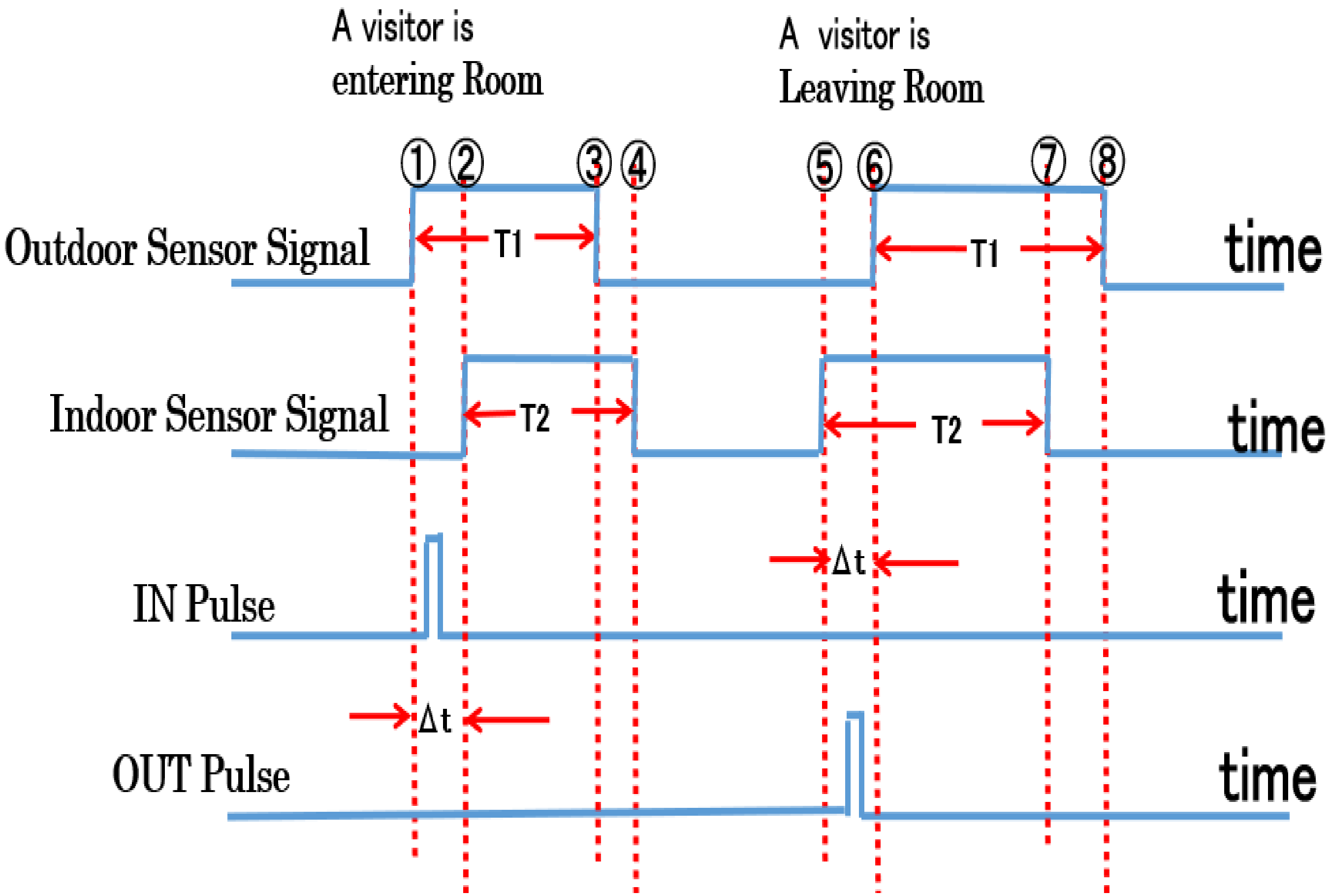
Compute $A[m] = \sum_{k=1}^{8} (1/8)\, H[m][k]\, a[k]$; Then we have $a(t) = \sum_{m=1}^{8} A[m]\, H_m(t)$;

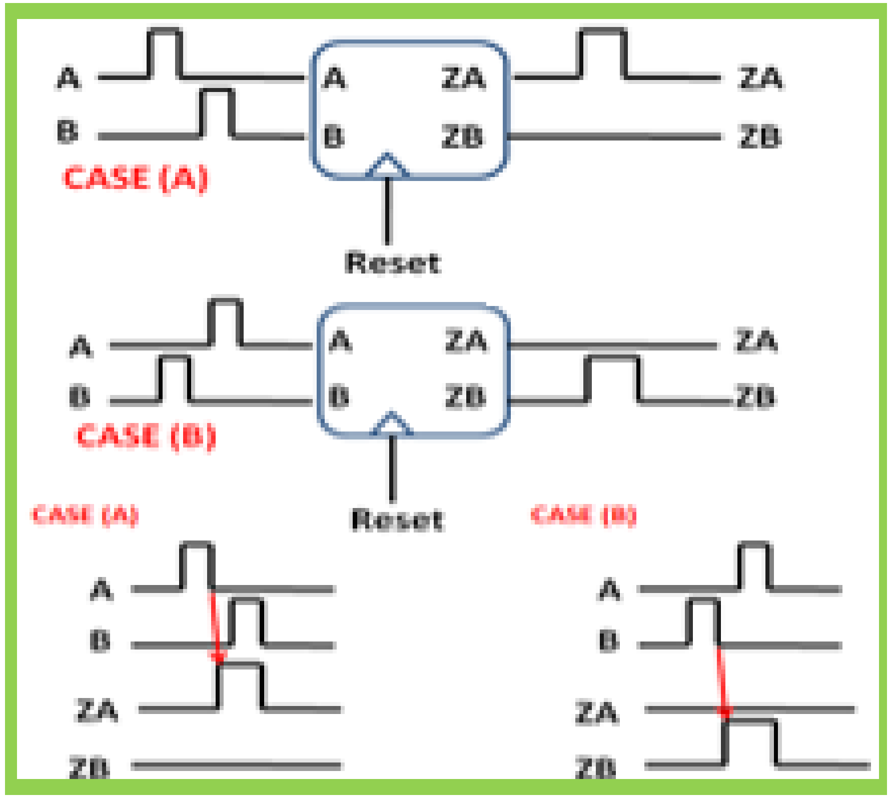
9 Artificial Intelligent Image Sensor


Visitor Counter System
Cd S Photo Cell IR sensor system block
Vdd
clock Generator
CLK
RESET
CLK
Power
PulseGen(1)
PulseGen(2)
Pulse1
Pulse2
Entry Pulse Judge Circuit(1)
X1
X2
Counter(1)
Counter(2)
Y1
Y2
CLK
Coming in / Going out Recoding System
Z
MAX
SUB(1)
Z
MaxAlert(1)
a2
a1
IRsensor(2)
IRsensor(1)
Power for Display
CLK
Y1
Y2
Z
OK
DisplayOut(1) 2021.4.1 07:04:00 p.m.
OK
Number of Total Visitors ... 1948
Current Visitors in the room... 74
AI camara watching People
Two Cd S Photo Cell IR Sensor System
Judge (1)
①IR
②IR
Counter System for People walking on the street
AI camara watching People

9 Artificial Intelligent Image Sensor

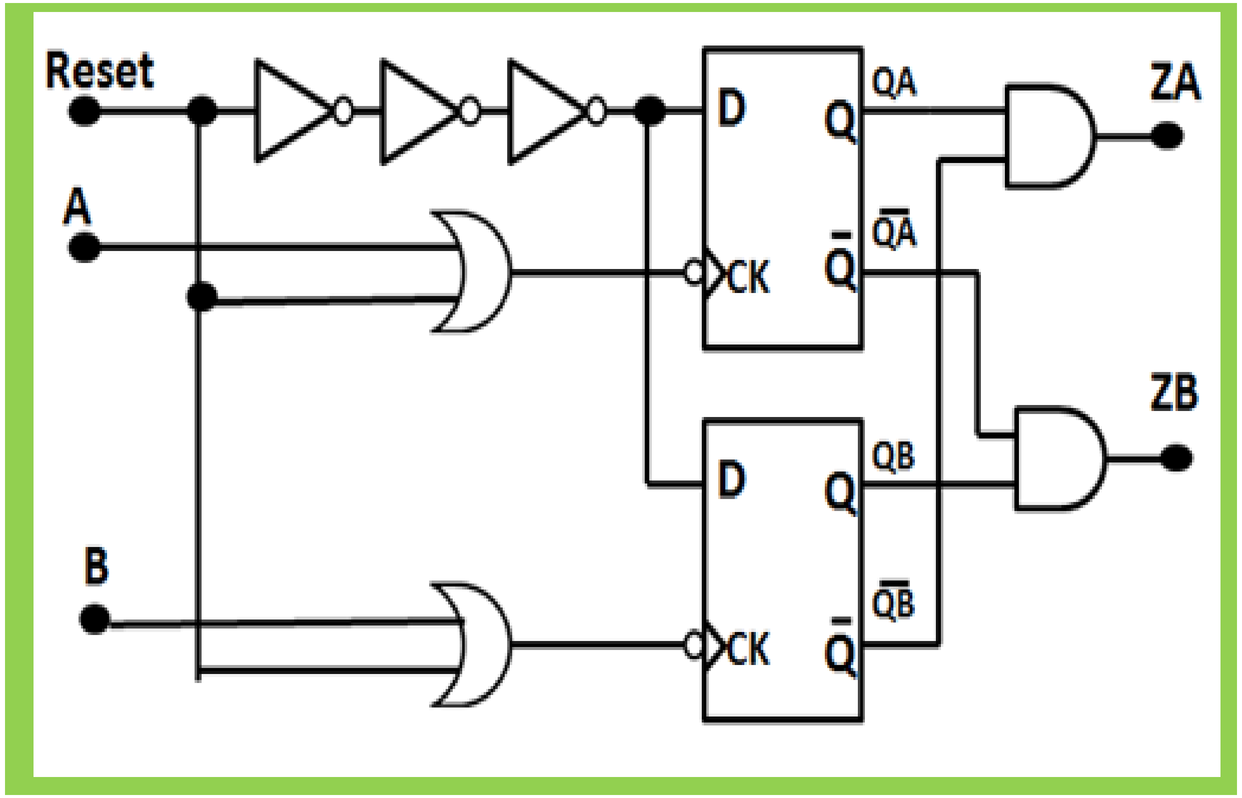
Digital Circuit Design for Visitor Counter System

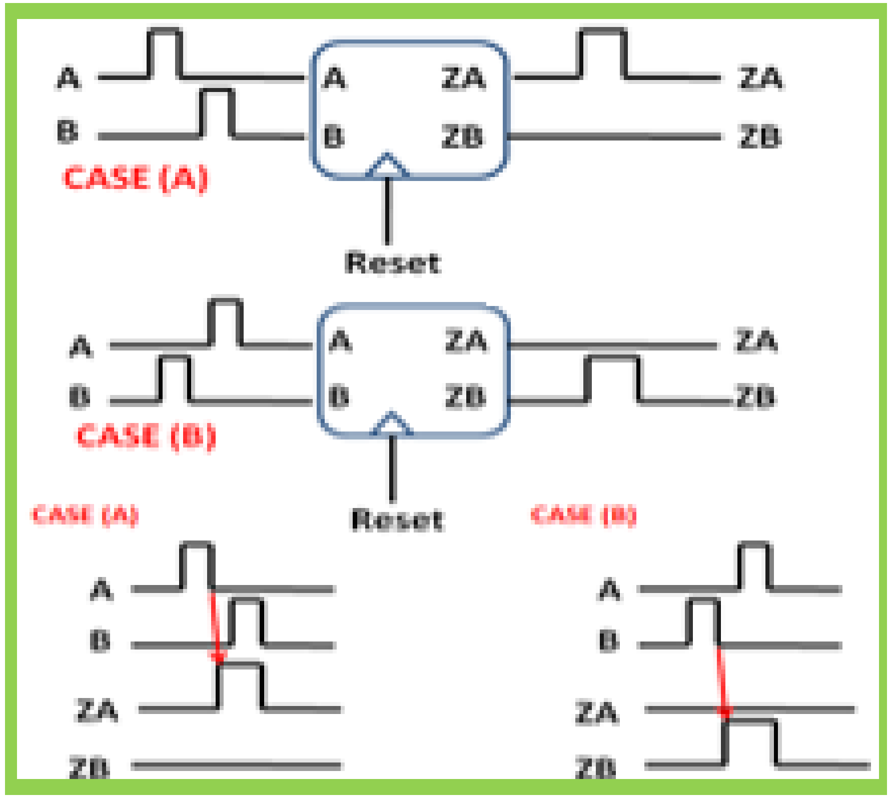




9 Artificial Intelligent Image Sensor

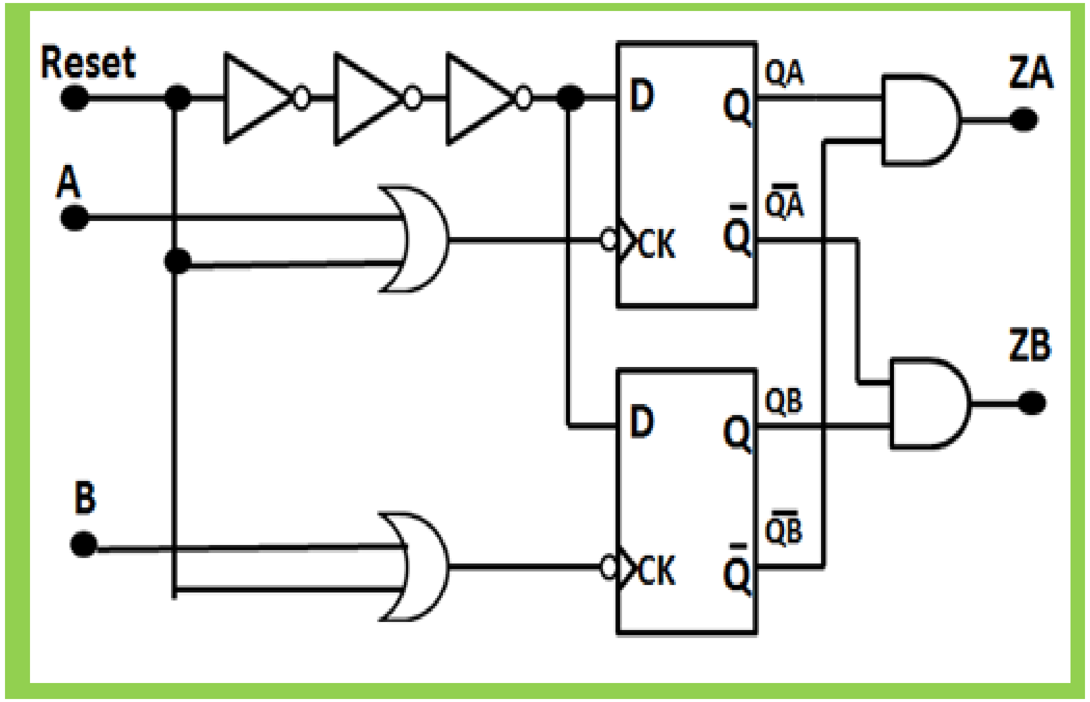
Digital Circuit Design for Visitor Counter System

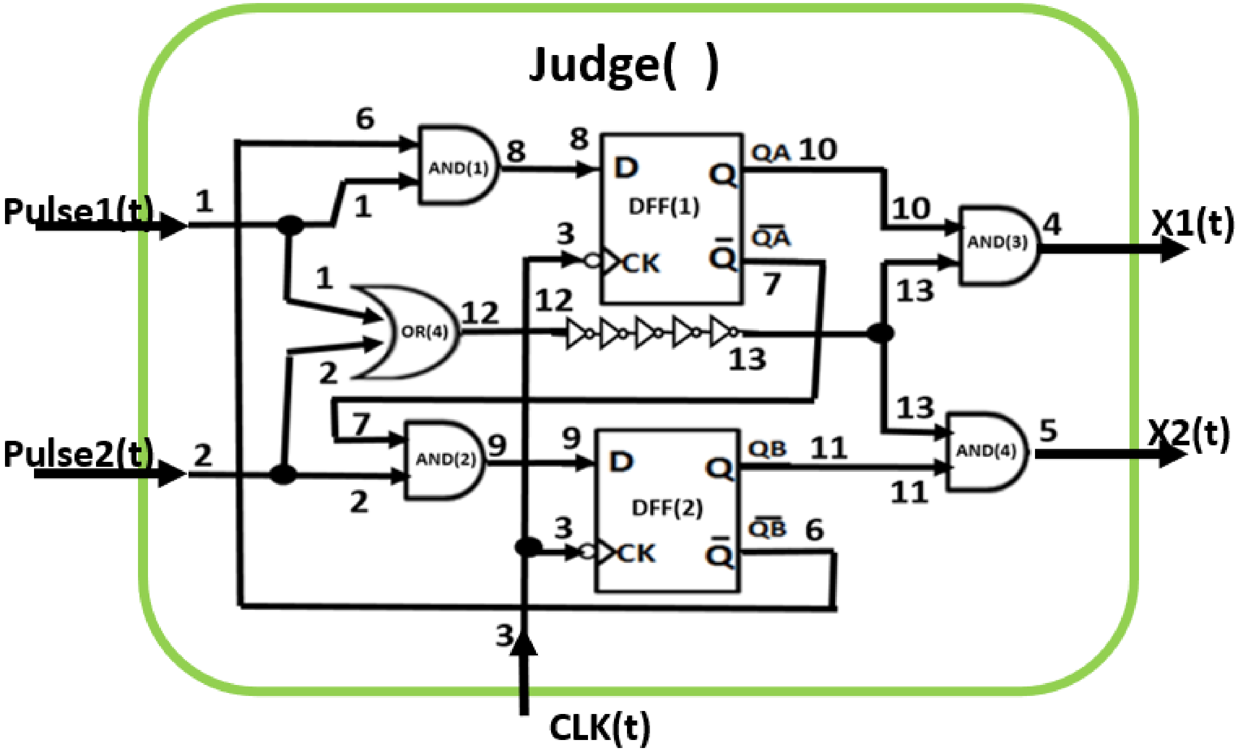




9 Artificial Intelligent Image Sensor

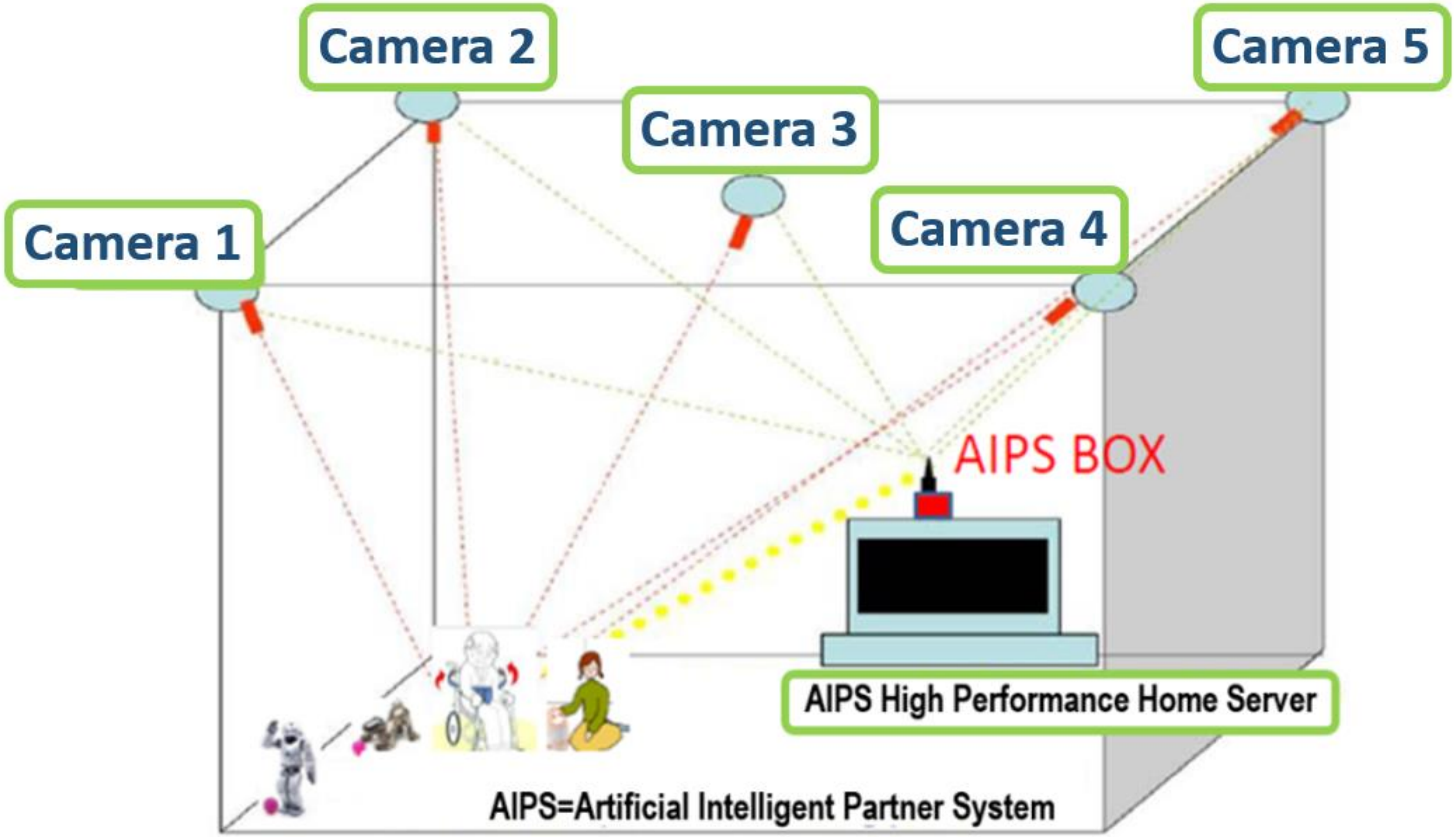
Digital Circuit Design for Visitor Counter System



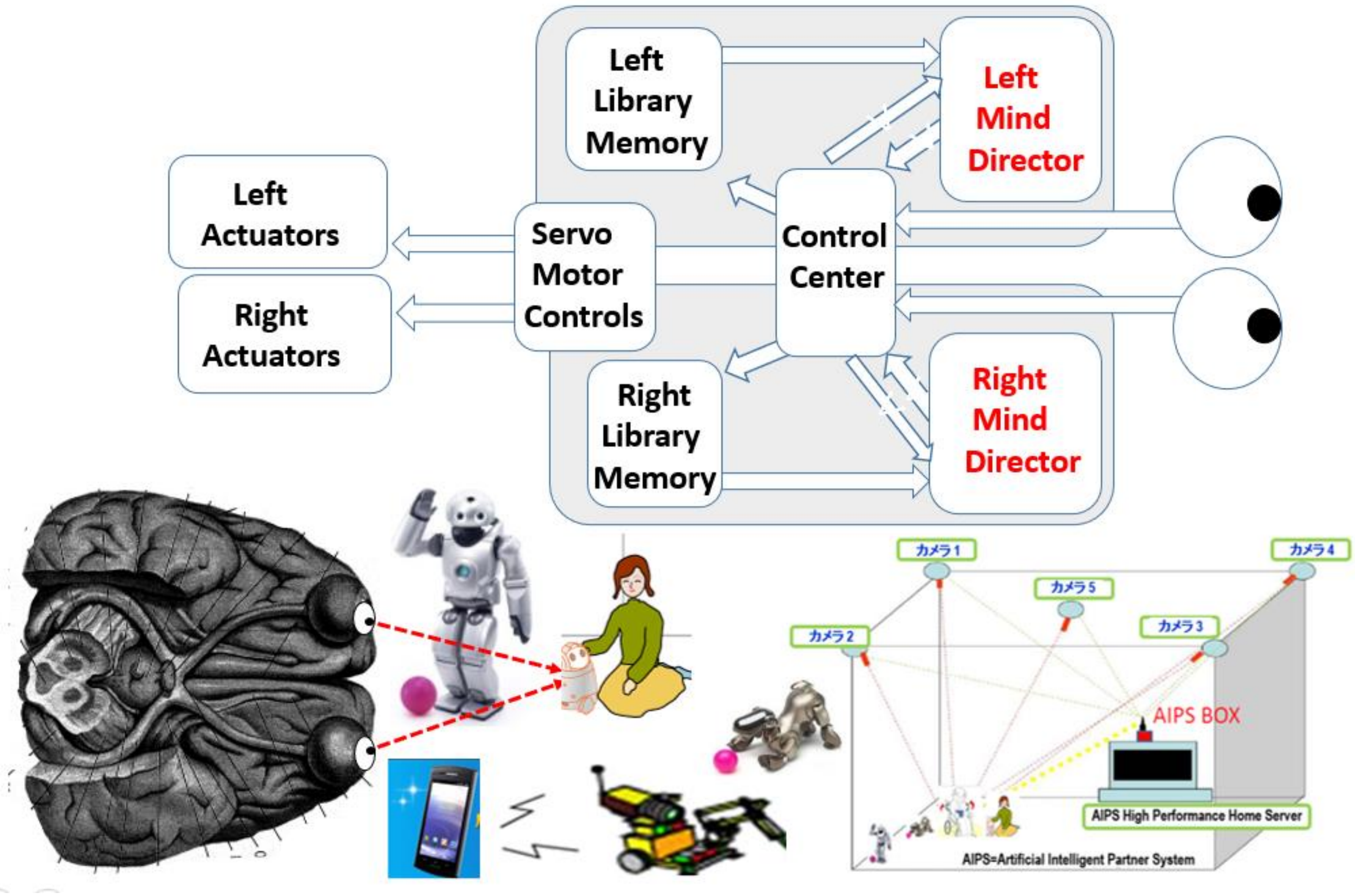


9 Artificial Intelligent Image Sensor

Digital Circuit Design for Home Care System


Camera 2
Camera 5
Camera 3
Camera 1
Camera 4
AIPS BOX
AIPS High Performance Home Server
AIPS=Artificial Intelligent Partner System

9 Artificial Intelligent Image Sensor



Left Library Memory

Left Mind Director

Left Actuators

Servo Motor Controls

Control Center

Right Actuators

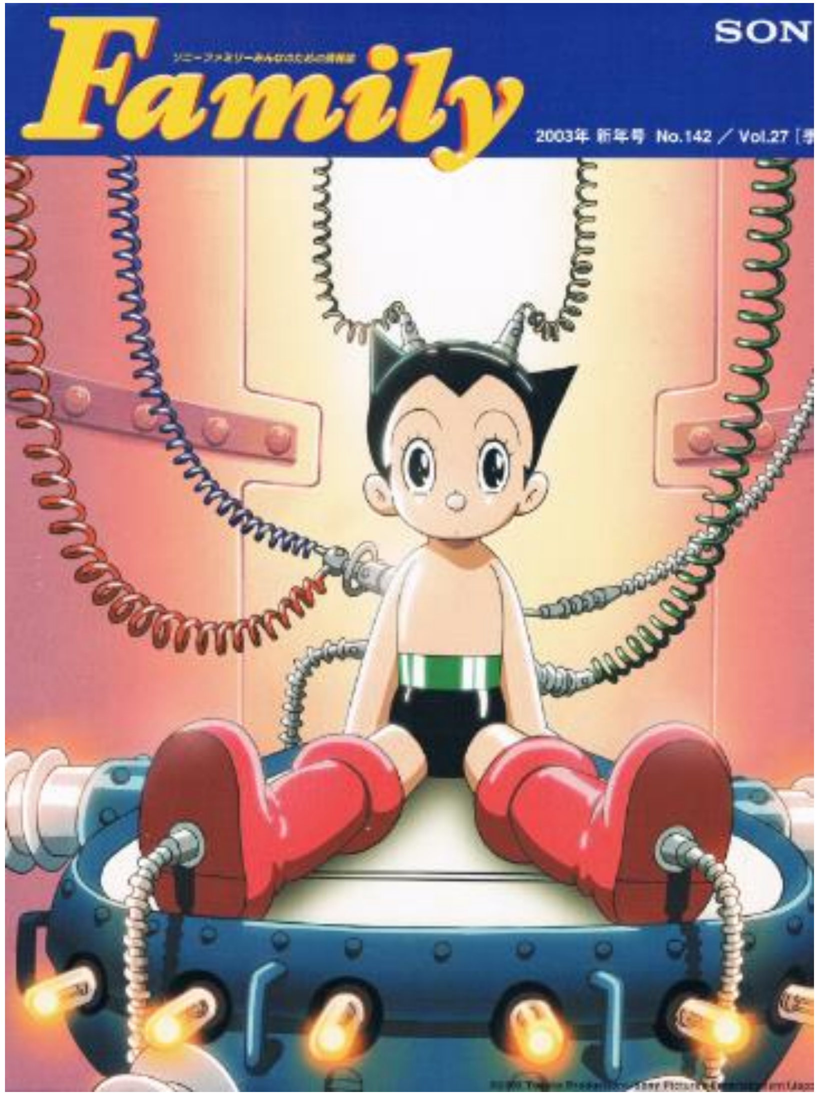
Right Library Memory

Right Mind Director

カメラ1
カメラ4
カメラ5
カメラ2
カメラ3
AIPS BOX
AIPS High Performance Home Server
AIPS=Artificial Intelligent Partner System

9 Artificial Intelligent Image Sensor

Artificial Intelligent Image Sensors use many dedicated real time hardware engines. To direct an image sensor pin-pointed to the right object quickly, a servomotor controlled digital feedback system is desired and being built with many tools including high performance CMOS digital circuits and clever software designs.


Family
ソニーファミリーみんなのための情報誌
SON
2003年 新年号 No.142 ／ Vol.27 [

AIBO Model ERS-210

AIBO Model SDR-3