



Design and FPGA Implementation of a Novel Radar Adaptive Post Detection Integration Algorithm

{F.M. Ahmed, H. Kamel, A. Fouad, and A. Fahmy}*

Abstract: In the present work, design and Field Programmable Gate Array (FPGA) implementation of a new Adaptive Post detection Integration (API) algorithm, designated as Conditioned Adaptive Post detection Integration (C-API), is proposed. The proposed C-API algorithm overcomes the problem of azimuth resolution degradation in the traditional API, especially for high signal to noise ratios (SNRs), and gives a robust performance against asynchronous pulse interference without affecting the detection capability of the traditional API. Computer simulations and experimental measurements are provided to validate the superiority of the proposed C-API algorithm against the traditional API and the Adaptive Binary Integrator (ABI).

1. Introduction

In adaptive radar detection, one way to increase the detection capability is to use the post detection integration for the received returns during sweeps. The final decision about the presence of a target is made by comparing the integrated signal to an adaptive threshold [1]. Adaptive thresholding is achieved by scaling the noise level estimate with certain constant to obtain the required probability of false alarm, P_{fa} [2]. The resulting scheme is called Adaptive Post detection Integration (API). The probability of false alarm of the (API) is extremely sensitive to randomly arriving impulse interference which is undesirable in radar systems. The above problem can be overcome by using ABI which is simple form of pulse integration even if it is less efficient than ideal API [3].

Many references discussed the performance of the API and ABI detectors especially under non-homogeneous background and randomly arriving impulse interference [3-8]. The problem of azimuth resolution degradation which is presented by the API, especially at high SNRs, has not been exploited in any other literature. In this paper, a Conditioned Adaptive Post detection Integration (C-API) algorithm is presented to overcome the problem of azimuth resolution degradation while keeping the detection performance of the ideal API. The proposed C-API detector is also simple to design and implement using recent technologies like Field Programmable Gate Arrays (FPGAs) [7].

The rest of this paper is organized as follows; section 2 introduces the idea of the proposed C-API. Performance evaluation of the proposed C-API against the API and the ABI through radar Receiver Operating Characteristic (ROC) is introduced in section 3. The design and FPGA implementation of the proposed C-API is introduced in section 4. Finally, experimental results of the implemented hardware are demonstrated in section 5.

* Egyptian Armed Forces, Egypt.

2. The proposed C-API Algorithm

The simple functional block diagram describing the proposed C-API is shown in Fig.1. It consists of non-recursive delay line integrator followed by Constant False Alarm Rate (CFAR) algorithm. The direct, middle, and final tapes of the integrator delay line, designated as A , B , and C are used to give an estimate, E_s , for the input video signal strength at the autocorrelation edges at the output of the integration stage.

For the processing of N -pulses in one beam, the number of autocorrelation samples are $(2*N)-1$. Equation (1) is proposed to get an estimate, E_s , of the useful target video signal appears at the first $N/2$ and final $N/2$ samples of the output autocorrelation.

$$E_s = | | A - C | - B | \quad (1)$$

Assuming that the video signal at the output of the envelop detector and input to integrator is described by Equation (2).

$$V(n) = S(n) + \varphi(n) \quad (2)$$

where $S(n)$ is the useful target signal and is defined only at $n=M*L$, where M is integer ($0 \leq M < N$), and L is the number of samples during one pulse repetition period. Also, $\varphi(n)$ is the Additive White Gaussian Noise (AWGN) with mean, m , and variance, σ .

A , B , and C can be described by the following equations:

$$A(n) = \begin{cases} S(n) + \varphi(n), & 0 \leq n < L*N \\ \varphi(n), & O.W. \end{cases} \quad (3)$$

$$B(n) = \begin{cases} S(n) + \varphi(n) , & L*N/2 \leq n < (3/2)*L*N \\ \varphi(n) , & O.W. \end{cases} \quad (4)$$

$$C(n) = \begin{cases} S(n) + \varphi(n) , & L*N \leq n < 2*L*N \\ \varphi(n) , & O.W. \end{cases} \quad (5)$$

By substitution for A , B , and C of (3), (4), and (5) into (1), the required estimate, E_s , can be obtained:

$$E_s(n) = \begin{cases} S(n) + \varphi(n) , & 0 \leq n < L*N/2 \\ S(n) + \varphi(n) , & (3/2)*L*N \leq n < 2*L*N \\ \varphi(n) , & O.W. \end{cases} \quad (6)$$

By dividing E_s over the calculated average noise power obtained by the CFAR circuitry, we can get an estimate for the input SNR at the autocorrelation edges. If this SNR exceeds certain level k , obtained by simulation, at the same time of deciding a target by the CFAR circuit, the final decision should be considered as no target. By doing so, a trimming for the autocorrelation edges at the final output is achieved. However, Fig. 2 shows the different signals describing the processing in the proposed algorithm for $N=10$ pulses, $SNR=10$ dB.

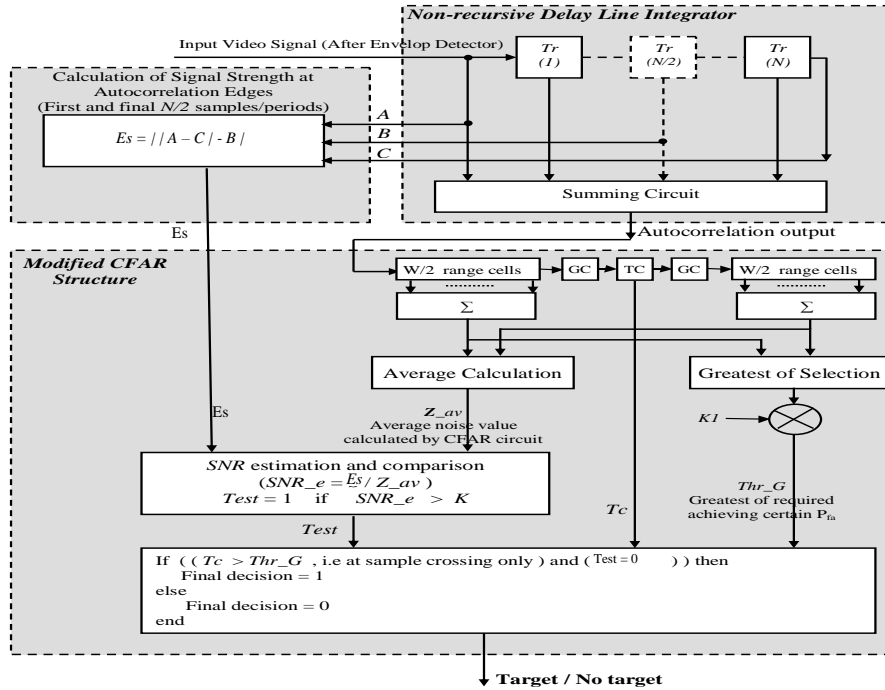


Fig.1 Simple functional block diagram describing the proposed C-API

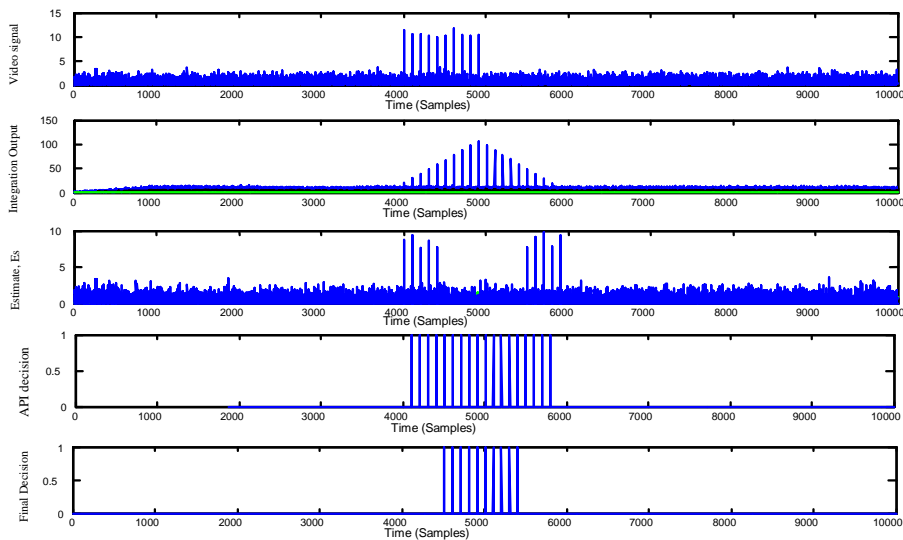


Fig.2 Different signals describing the processing in the proposed algorithm for $N=10$ pulses, $SNR=10$ dB.(a) Input video signal, (b) Integration autocorrelation output, (c) The calculated estimate, Es , (d) Traditional API output, and (e) The final output of the proposed C-API

2. Performance Evaluation

Computer experiments were performed to verify the superiority of the proposed algorithm. Using MATLAB package, the variance of the additive white Gaussian noise was fixed to unity while different SNRs were produced by scaling the signal amplitude. The number of pulses to be integrated, N , is 10 pulses. The number of samples in one pulse period is 1 sample, while the number of samples in one pulse repetition interval is 100 samples. The total sample size is 10,000. The designed probability of false alarm, P_{fa} , is chosen to be 10^{-5} . The total CFAR window length is chosen to be 16 samples. Results for detection were averaged over 10,000 independent trials.

Detection curves comparison is shown in Fig. 3. It is clear that the detection performance of the proposed C-API is identical to that of the API outperforming the detection performance of the ABI. This means that the proposed algorithm does not affect the detection performance of the API.

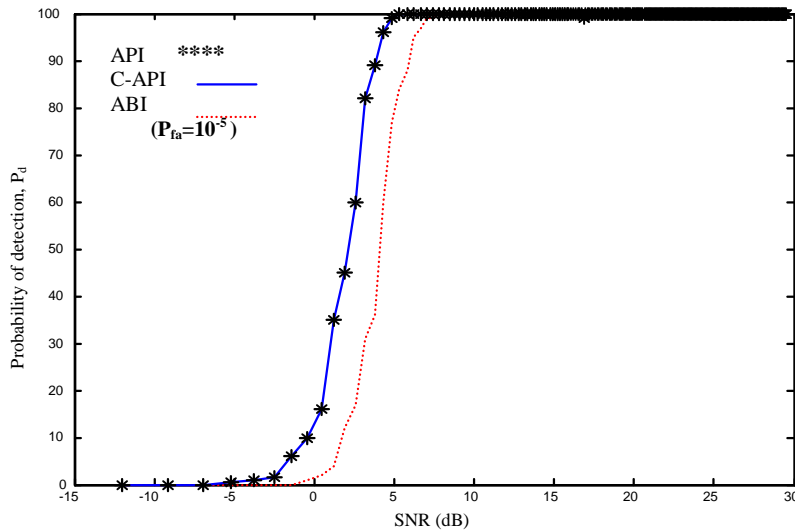


Fig. 3 Comparison between the detection curves for the API, ABI, and the proposed C-API.

Comparisons between the azimuth resolution performances of the mentioned algorithms for single target, two adjacent targets, and two separated targets (one beam separation) are shown in Figs. 4, 5, and 6 respectively.

Fig. 4 shows the superiority of the proposed C-API to keep the azimuth resolution of detecting one target similar to that of the ABI and outperforming that of the API.

The superiority of the proposed C-API in resolving two adjacent targets over that of the API or ABI is shown in Fig. 5. Although the number of samples representing each target are reduced to the half of its nominal value, but each target still can be detected and resolved.

For two separated targets with one beam separation, as shown in Fig. 6, the proposed C-API succeeded to resolve both targets like the ABI and outperforming the API. In general, the proposed C-API succeeded in resolving the two targets in all cases outperforming that of the API.

The effect of asynchronous interference on the probability of false alarm at different Interference to Noise Ratios (INRs) for the mentioned algorithms is shown in Fig. 7. It is clear that the probabilities of false alarm for the three detectors are almost the same for different INRs.

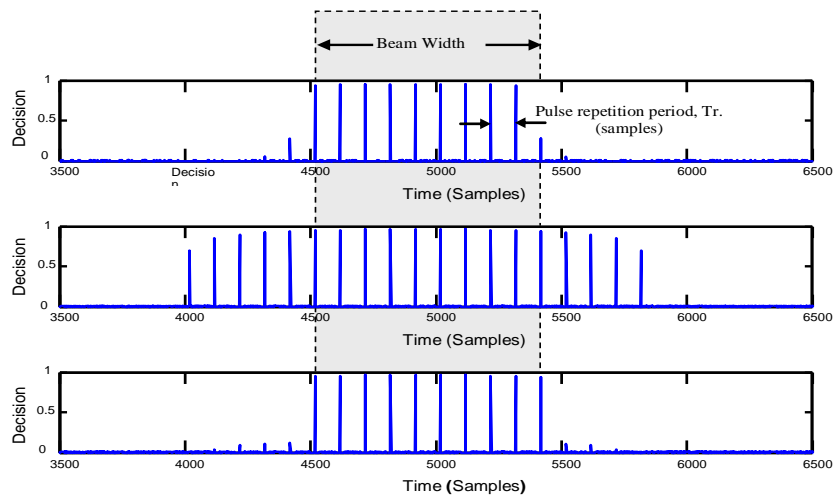


Figure 4. Azimuth resolution performance of detecting one target averaged over different SNRs (0-30 dB) for ABI (top), traditional API (middle), and the proposed C-API (down).

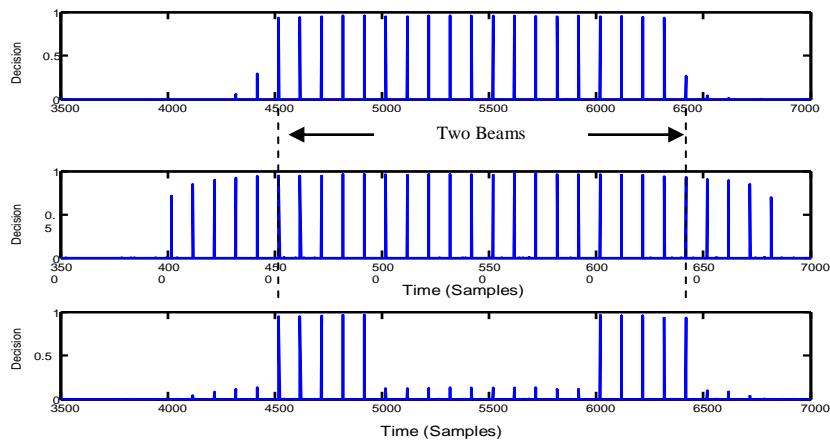


Figure 5. Azimuth resolution performance of detecting two adjacent targets averaged over different SNRs (0-30 dB) for ABI (top), traditional API (middle), and the proposed C-API (down).

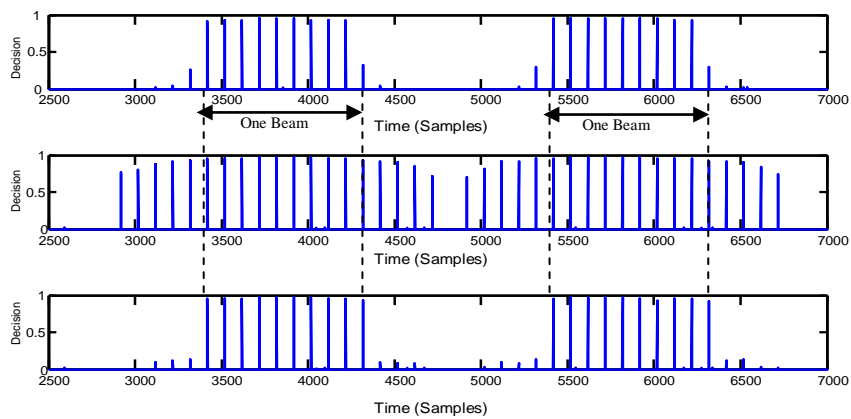


Figure 6. Azimuth resolution performance for detecting two separated target (one beam separation) averaged over different SNRs (0-30 dB) for ABI (top), traditional API (middle), and the proposed C-API (down).

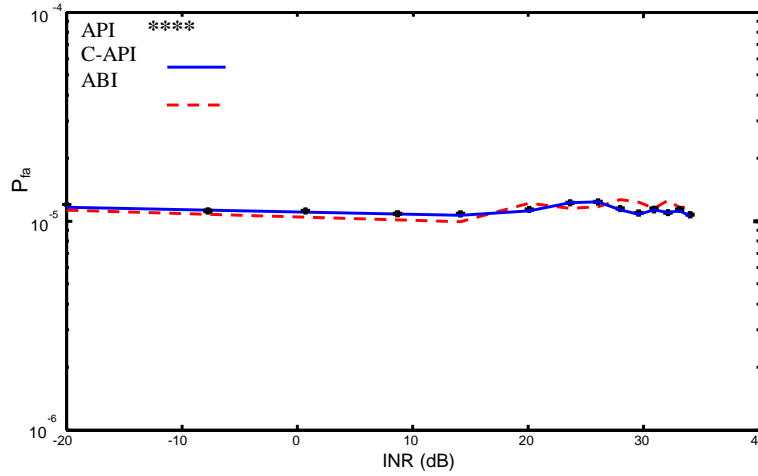


Fig. 7 The effect of asynchronous interference on false alarm probabilities of the API, ABI, and the proposed C-API at different INRs.

Design and Implementation of the Proposed C-API Using FPGA

The proposed C-API is designed and implemented to fit with Lab Volt laboratory radar system with the following video signal parameters [9]:

- The pulse width is 0.25 ms,
- The number of pulses to be integrated is 10 pulses,
- The pulse repetition period is 7 ms,
- The maximum amplitude of the video signal is 5 V,
- The simulated noise is an Additive White Gaussian Noise (AWGN) with zero mean and adjustable variance,

The proposed C-API is designed and implemented using FPGAs which have become one of the prevailing technologies for fast prototyping and implementation of complex digital systems [10, 11]. The general block diagram of the designed C-API is shown in Fig.8. The output analog video signal from the radar is converted into digital form with a sampling rate of 16 kHz, which is chosen according to Shannon sampling theory. The output from the analog to digital converter is fed to the used FPGA chip. The word length is chosen to be 8 bits. The (GO) CFAR window (W) is chosen to be 32 samples.

The CFAR processor is designed to achieve different probabilities of false alarm by changing the value of the detector multiplication factor, K. For the purpose of synchronization, the radar trigger is processed and fed back again to the radar display unit. The output decision of the proposed C-API is fed directly to the radar display unit. The C-API is designed and implemented based on FPGA using the Xilinx package ISE 9.1i with the ModelSim 6.0 simulator. The design was downloaded on the Spartan3-200k starter kit [12]. However, results of ModelSim simulation for different cases of radar signal situation are shown in Fig. 9 to Fig. 11. Simulation results give the same results obtained by Matlab simulations.

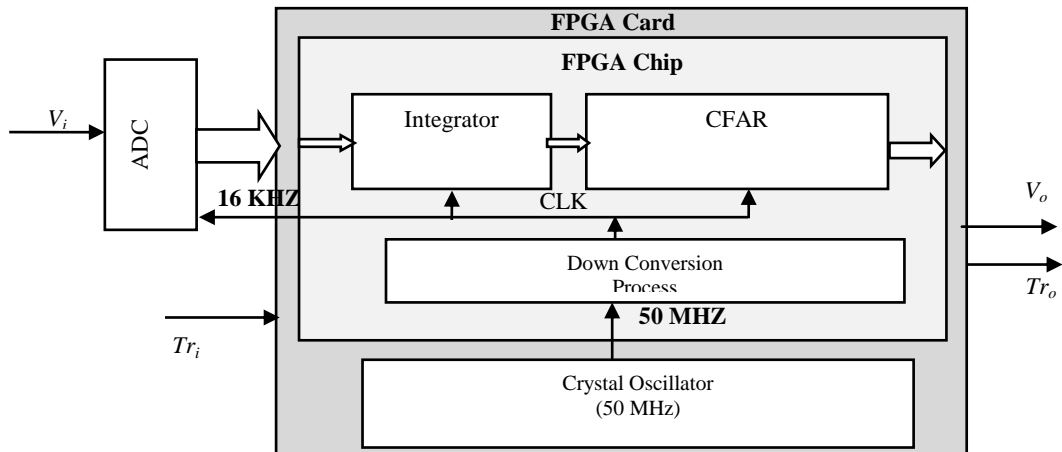


Fig. 8 The general block diagram of the designed C-API

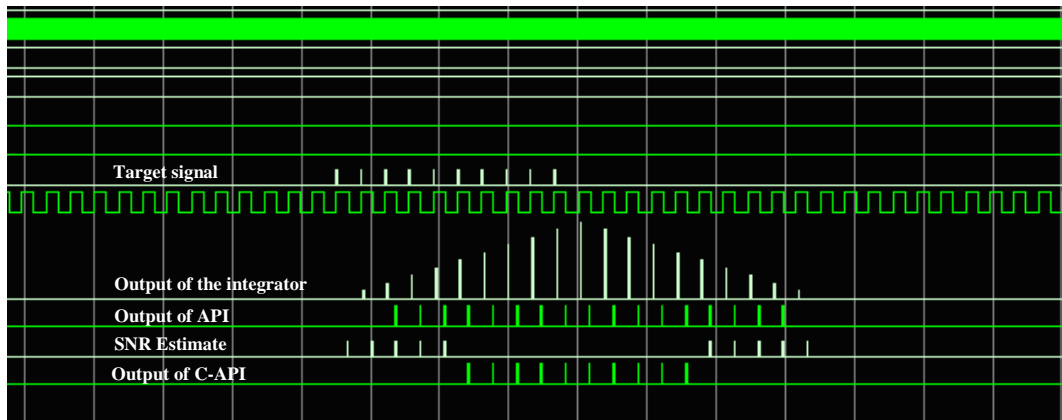


Fig. 9 ModelSim simulation of the proposed C-API in case of single target

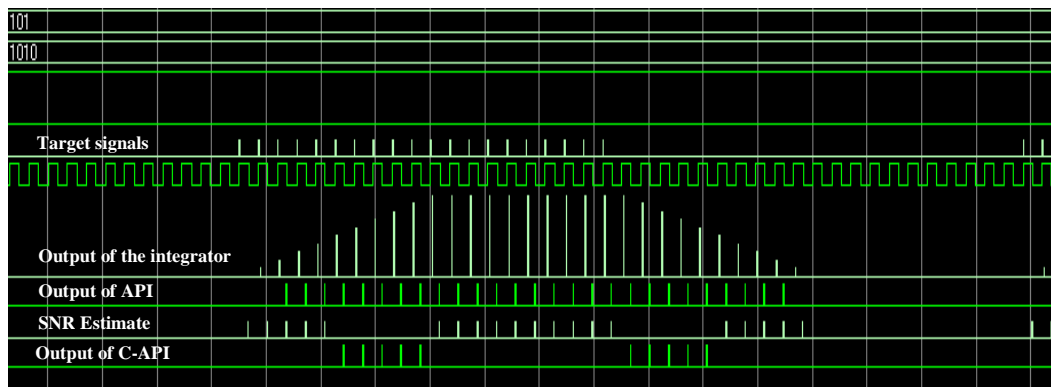


Fig. 10 ModelSim simulation of the proposed C-API in case of two adjacent targets

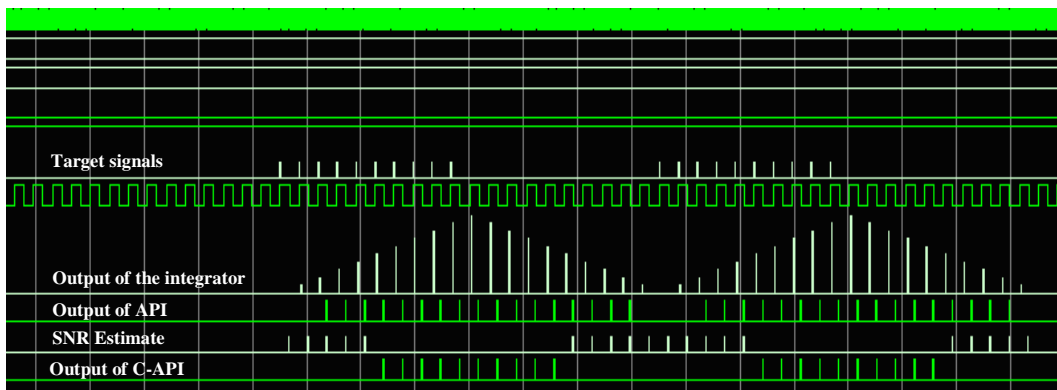


Fig. 11 ModelSim simulation of the proposed C-API in case of two close targets (one beam separation)

Experimental Results

The experimental setup used to test the performance of the proposed C-API is shown in figure 12. Two FPGA kits are used, one for simulating Lab Volt radar signals, and the other is used for implementing the proposed C-API. The radar signal simulator kit produces three cases of target situations. The first case is one target in one beam, the second case is two adjacent targets in two successive beams, and the last case is two targets in different beams separated by only one azimuth beam. However, results of experimental testing are shown in figure 13.

This figure is divided into two sections; the left one represents the simulated radar signal in different cases and the output of the traditional API, while the right section represents the same cases but with the output of the proposed C-API. Results showed an identical agreement between both Matlab or ModelSim simulations with the experimental results.

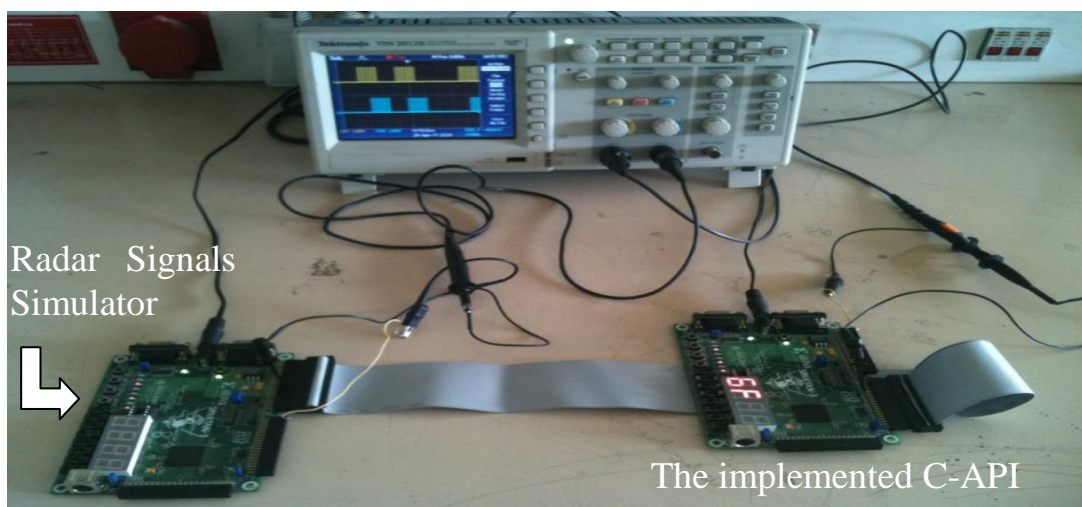
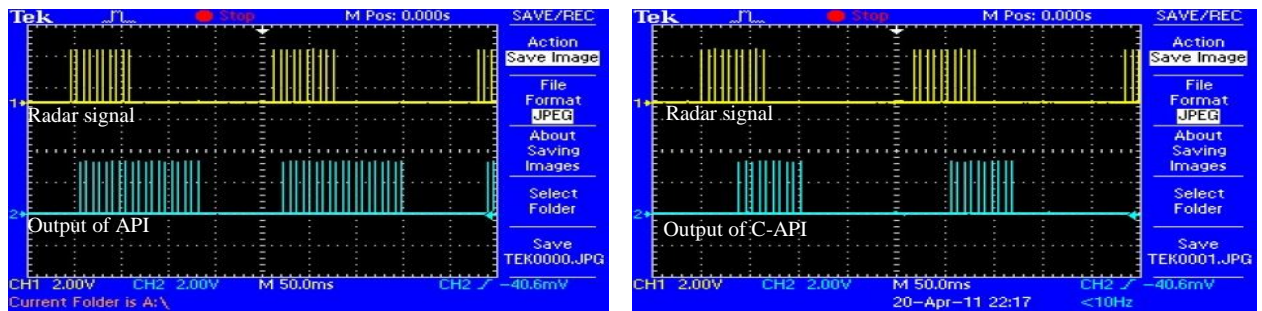


Fig. 12 The experimental setup used to test the performance of the proposed C-API

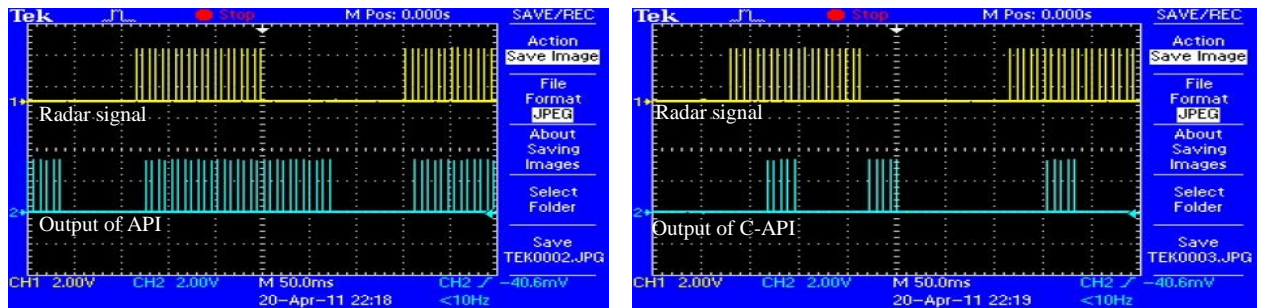
Conclusion

A new Adaptive Post detection Integration (API) algorithm, designated as Conditioned Adaptive Post detection Integration (C-API), is proposed, analyzed, and implemented with FPGA. The proposed C-API algorithm overcomes the problem of azimuth resolution degradation in the traditional API, especially for high SNRs, with a robust performance against asynchronous pulse interference while keeping the same detection performance of the API.

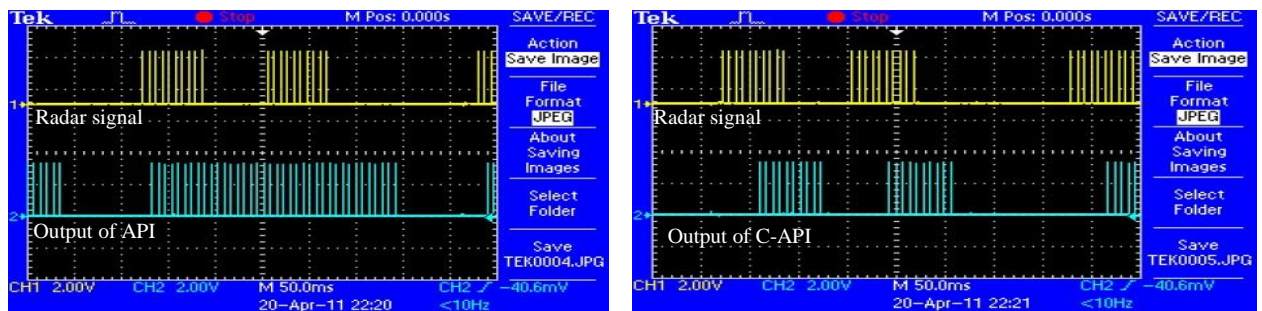
The proposed C-API succeeded to detect a single target with the same detection performance of the API and a better azimuth resolution performance. It also succeeded to detect and resolve adjacent targets outperforming both API and ABI. For closely separated targets, the proposed C-API achieved a better azimuth resolution performance than the API (like the ABI but with a better detection performance). The effect of the asynchronous pulse interference on the probabilities of false alarm for the mentioned three algorithms at different INRs is almost the same. Computer simulations and experimental results are provided to validate the superiority of the proposed C-API algorithm against the API and the ABI.



(a) One target in one azimuth beam



(b) Two targets in two adjacent azimuth beams



(c) Two targets in two azimuth beams separated by one azimuth beam

Fig. 13. Experimental results of the proposed C-API against the traditional API (The left figures for the API, and the right figures for the proposed C-API) for different target cases.

References

- [1] Difranto J.V. and Rubin W.L., "Radar Detection", Artech House, Norwood, Massachussets, 1980.
- [2] Finn H.M. and Johnson R.S., "Adaptive Detection Mode with Threshold Control as a Function of Spatially Sampled Clutter Estimates", RCA Review, 29, (3), pp. 414- 464, 1968
- [3] Levanon N., "Censored Video Integration in Radar Detection", Proc. 1990 IEEE Int. Radar Conf., pp, 511-513, 1990
- [4] M. Skolnik, "Introduction to Radar Systems, Third Edition", Mc Graw, 2002.
- [5] Shnidman, D.A., "Binary Integration for Swerling Target Fluctuations", IEEE Transactions on Aerospace and Electronic Systems, 34, (3), pp. 1043–1053, 1998.
- [6] X.W. Meng., "Performance analysis of ordered-statistic greatest of-constant false alarm rate with binary integration for M-sweeps", IET Radar Sonar Navig 2010, Vol. 4, Iss. 1, pp. 37–48, 2010.
- [7] Fathy M. Ahmed, K. H. Moustafa, A.Fouad, ,A.Fahmy "FPGA Based Design and Implementation of an Adaptive Binary Integrator", Proc. Of the 2010 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), pp. 409-419, Feb. 2011.
- [8] M.khalighi,M.hassan bastani, "Adaptive CFAR Processor For Non-homogenous Environment", IEEE transaction on Aerospace and Electronic System vol.36,no.3,july 2000
- [9] [Http://www.labvolt.com/products / telecommunications / radar / radar-training-system-8096](http://www.labvolt.com/products / telecommunications / radar / radar-training-system-8096).
- [10] Todman T.J., Constantinides G.A., Wilton S.J.E., Mencer O., LUK W., Cheung P.Y.K., "Reconfigurable computing: architectures and design methods", IEE Proc. Comput. Digit Tech., 152, (2), pp. 193–207, 2005.
- [11] A.M. Alsuwailam, S.A. Alshebeili, M.H. Alhowaish, and S.M. Qasim, "Field programmable gate array-based design and realization of automatic censored cell averaging constant false alarm rate detector based on ordered data variability", IET Circuits Devices Syst., Vol. 3, Iss. 1, pp. 12–21, 2009.
- [12] [Http : // www . xilinx . com / products / devkits /HW-SPAR3-SK-UNI-G.htm](http://www.xilinx.com/products/devkits/HW-SPAR3-SK-UNI-G.htm).