

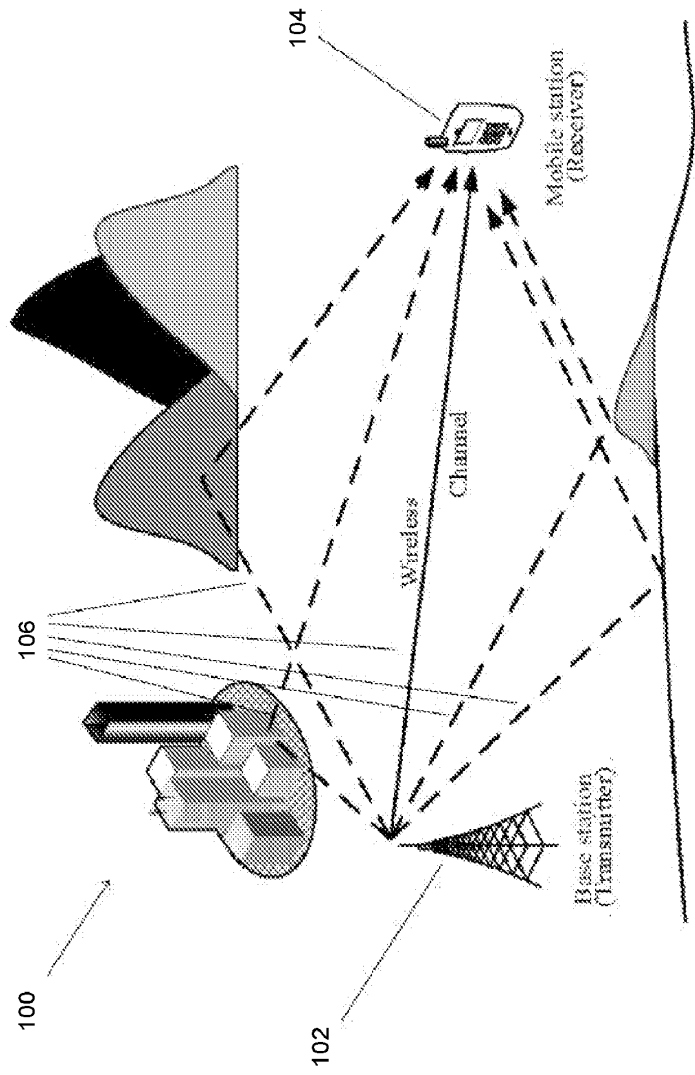


Fig. 1

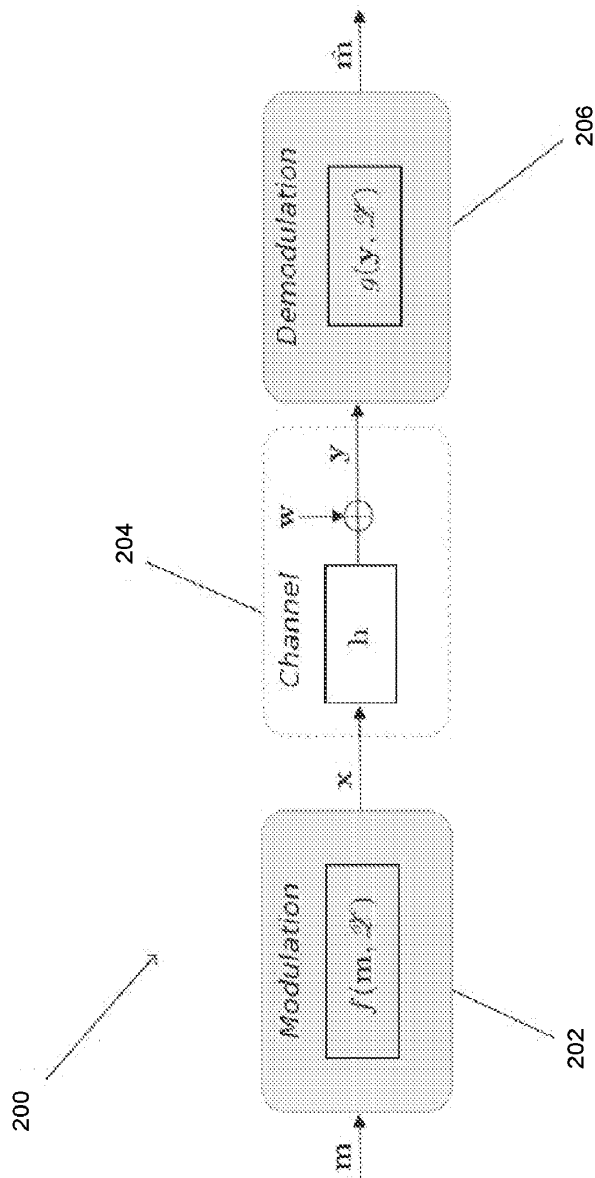


Fig. 2

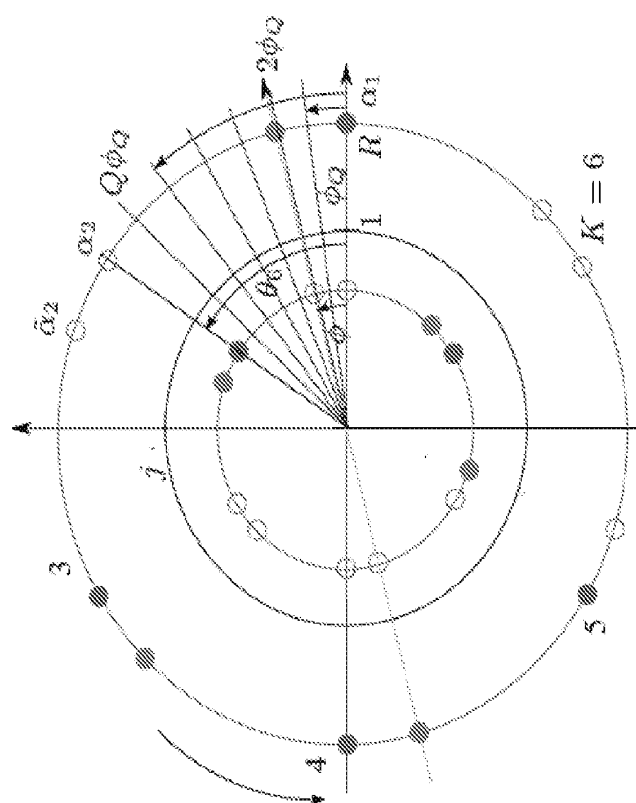


Fig. 3

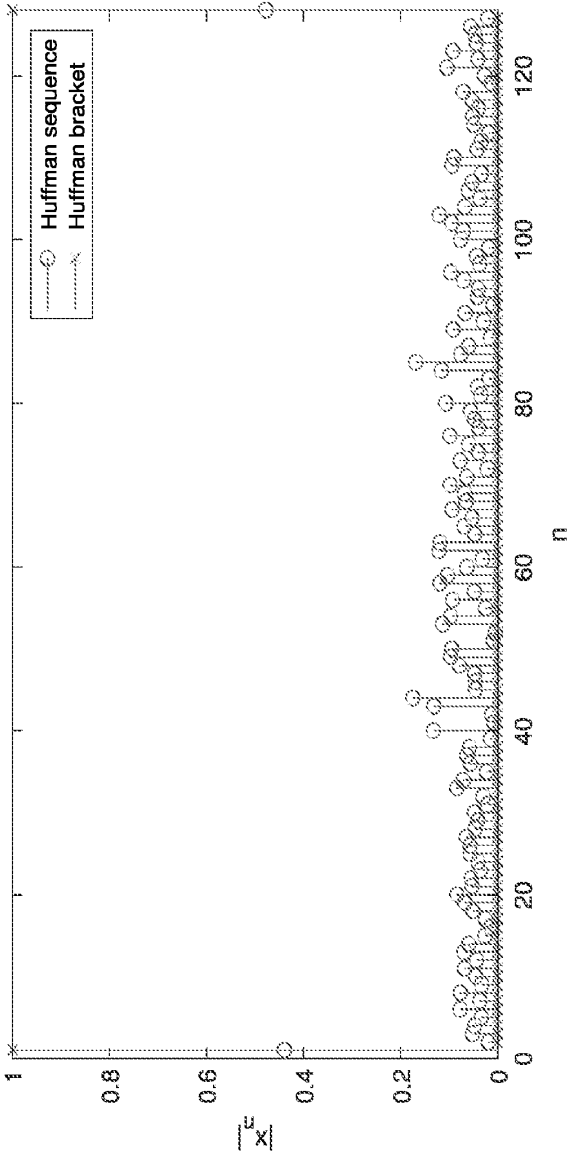
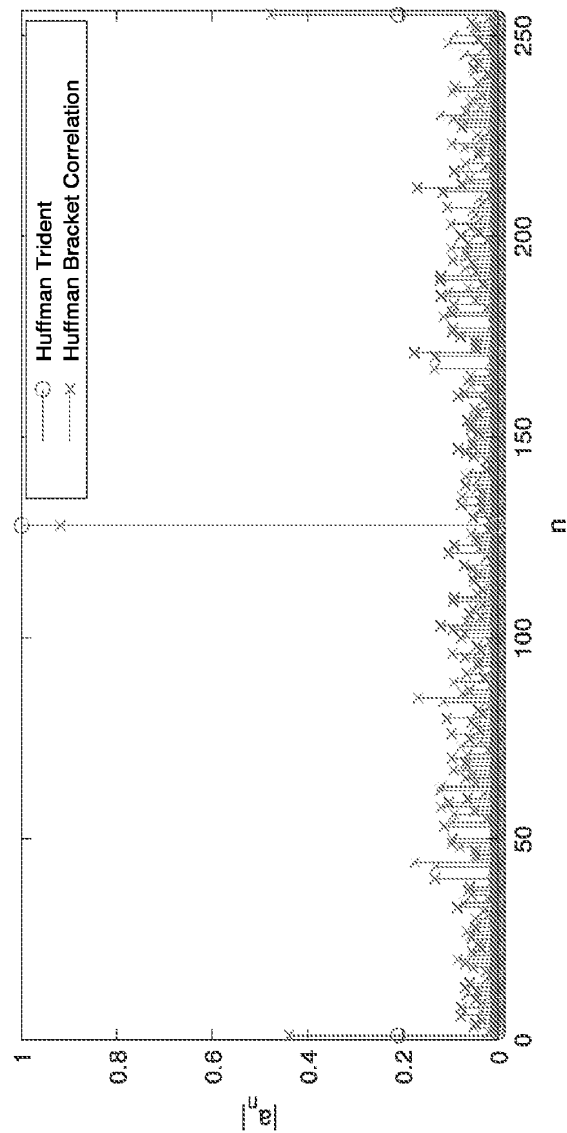


Fig. 4A



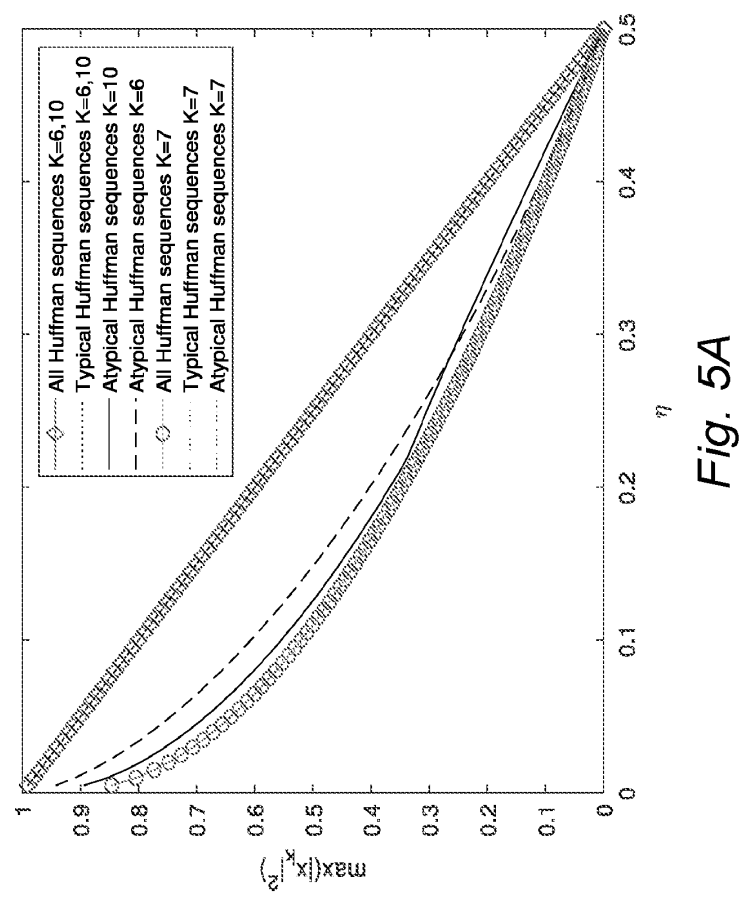


Fig. 5A

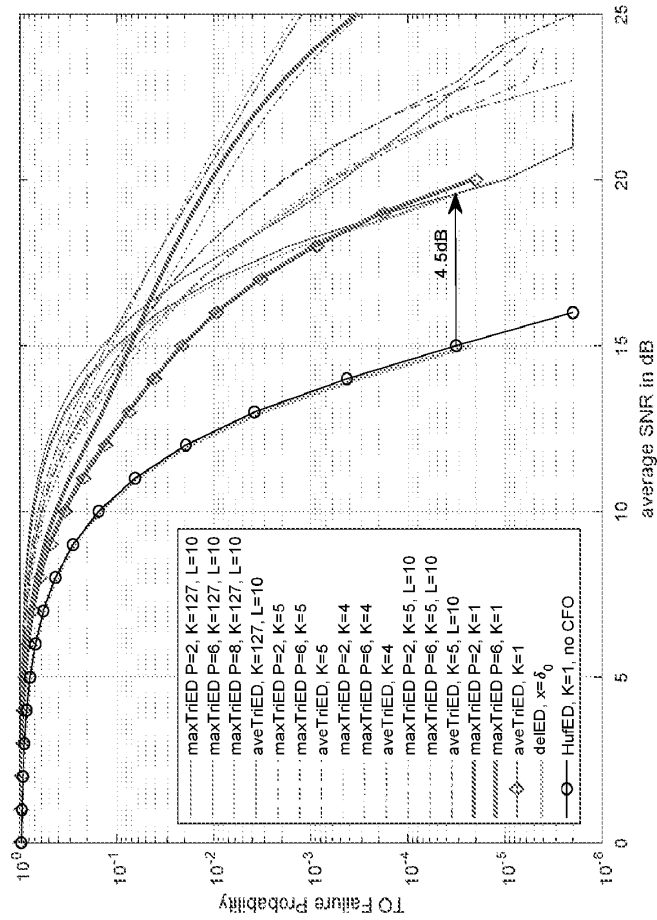


Fig. 5B

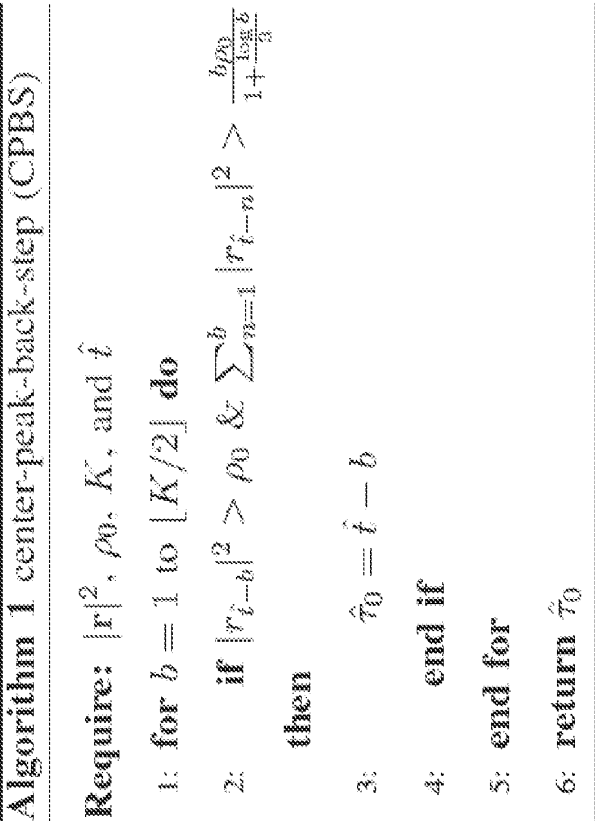


Fig. 6

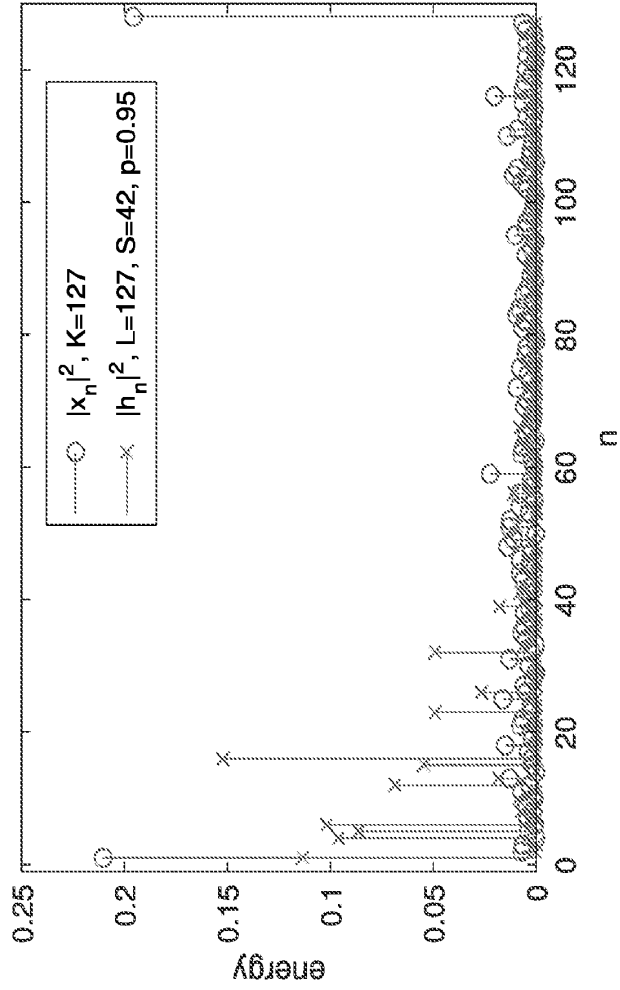


Fig. 7A

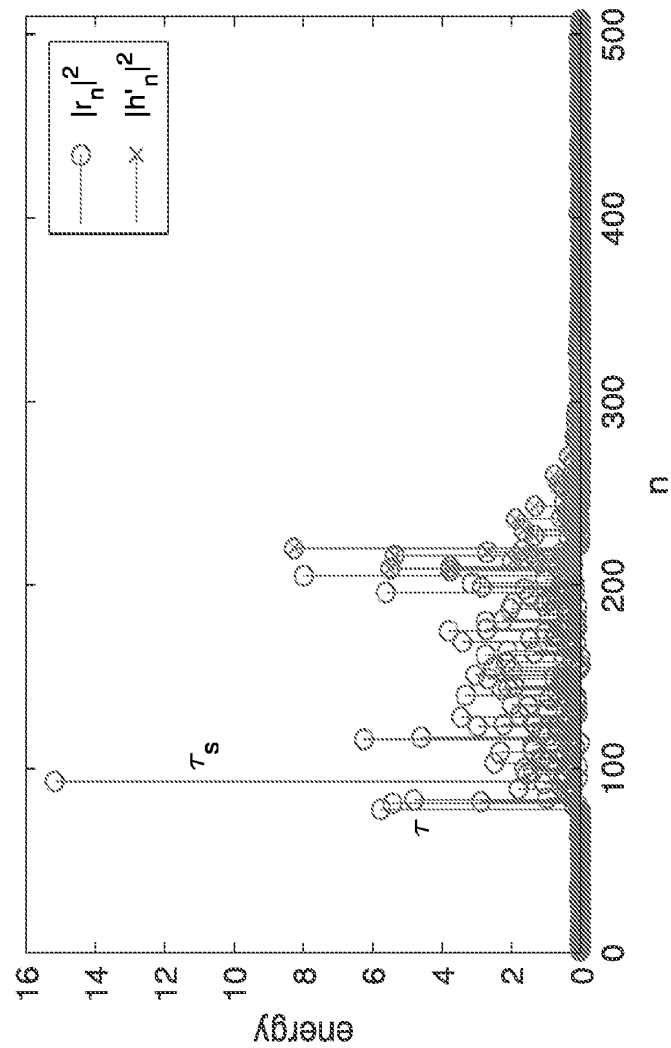


Fig. 7B

Algorithm 2 CIR energy-detection (CIREd)

Require: $\|\mathbf{r}\|^2$, $\hat{r}_0$, N_0 , K , L , and p

- 1: $E_r = \sum_{\ell=1}^L |r_{\hat{r}_0+K+1+\ell}|^2 - LN_0$;
- 2: $E_{\text{eff}} = |r_{\hat{r}_0+K+1}|^2$; $L_{\text{eff}} = 1$; $\mu = p^{LN_0/2Lr}$;
- 3: **while** $E_{\text{eff}} < \mu \cdot E_r$ **do**
- 4: $L_{\text{eff}} = L_{\text{eff}} + 1$;
- 5: $E_{\text{eff}} = E_{\text{eff}} + |r_{\hat{r}_0+K+L_{\text{eff}}}|^2$;
- 6: **end while**
- 7: **return** L_{eff}

Fig. 8

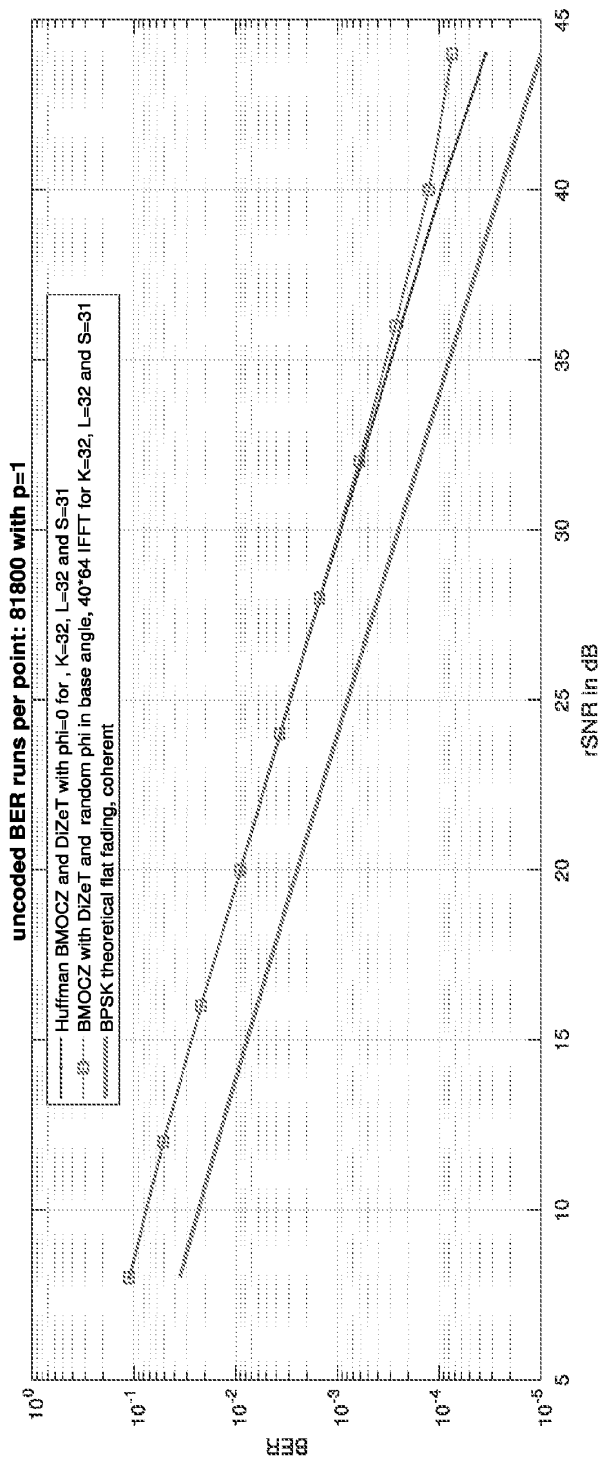


Fig. 9

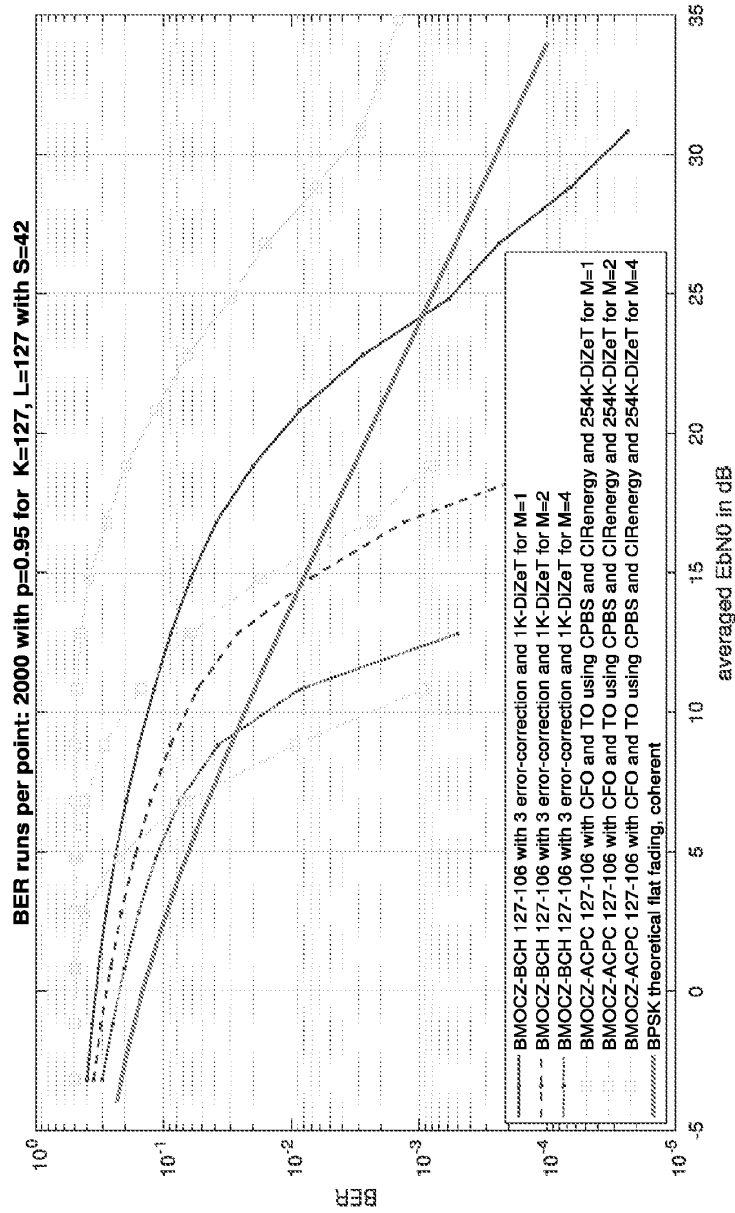


Fig. 10A

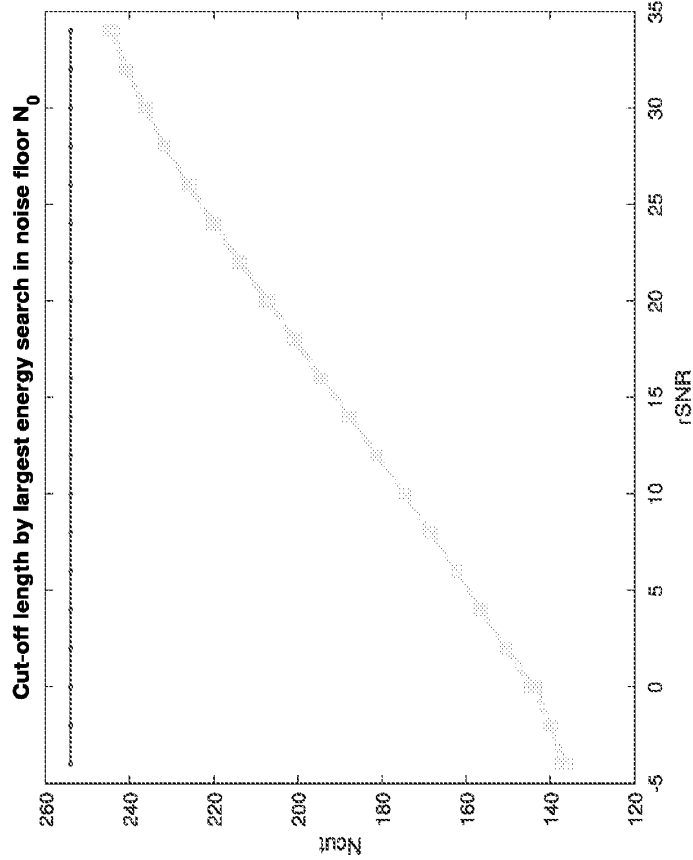


Fig. 10B

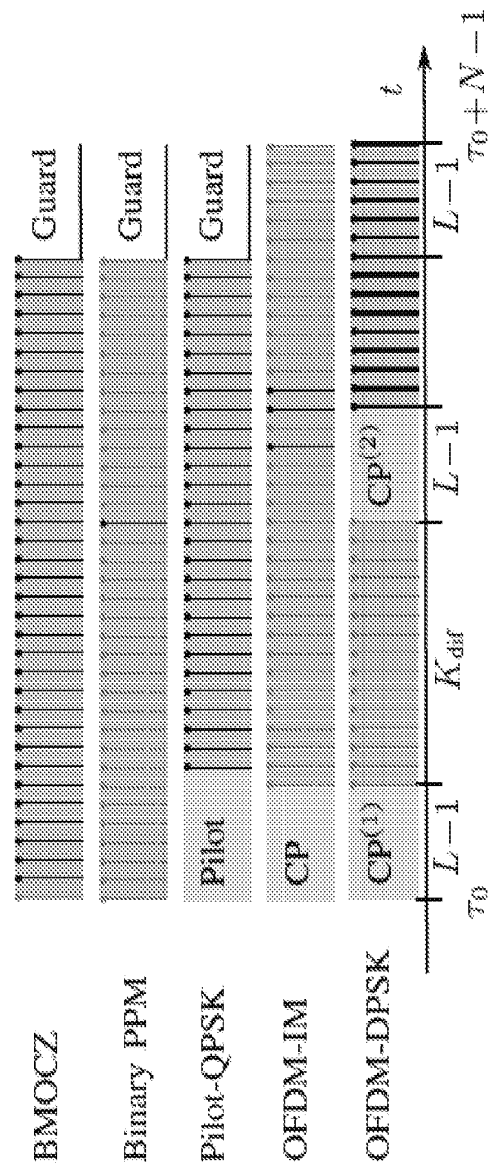


Fig. 11

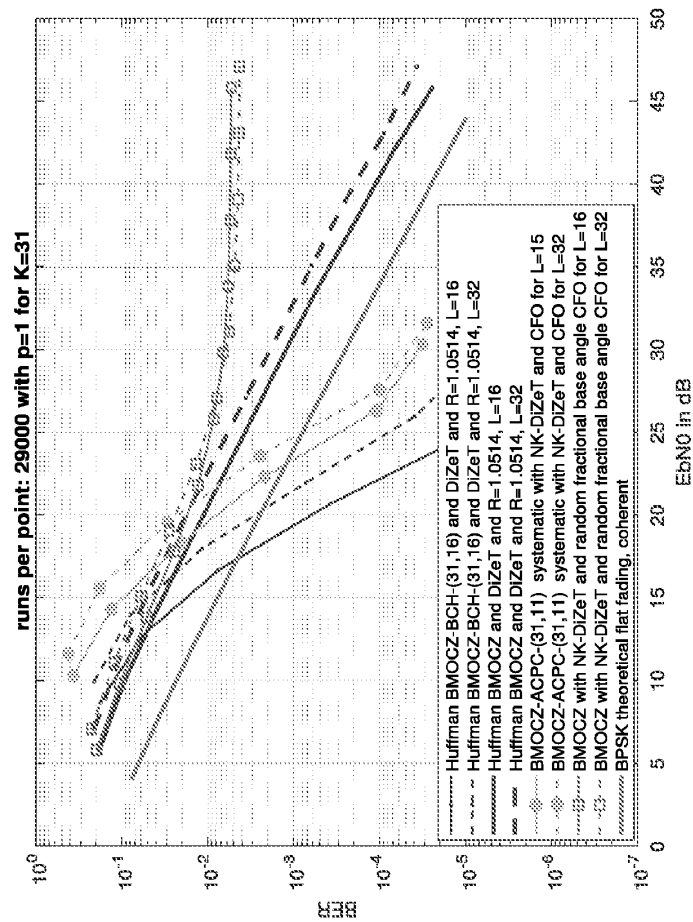


Fig. 12A

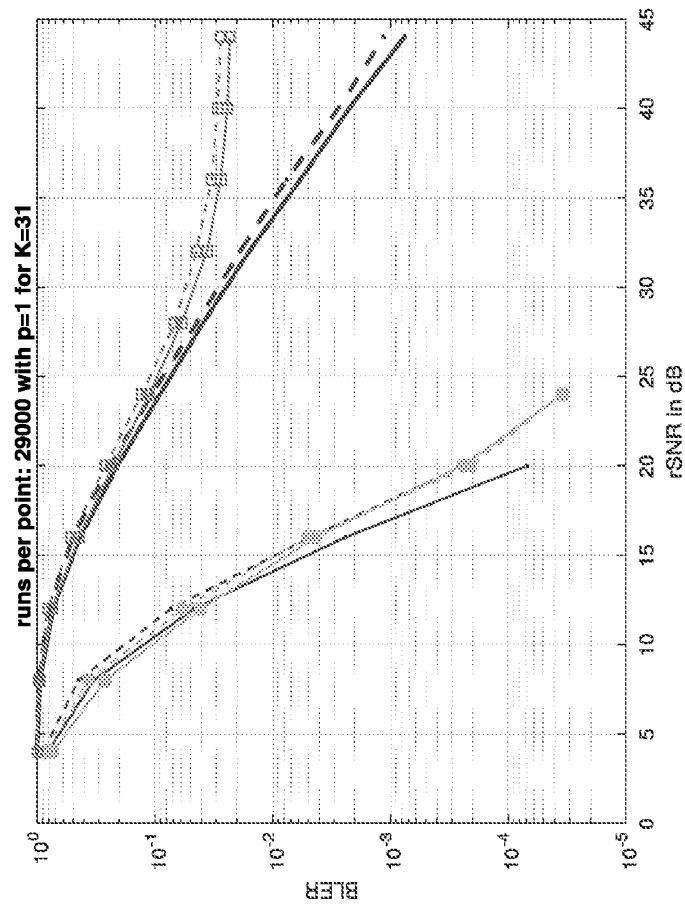


Fig. 12B

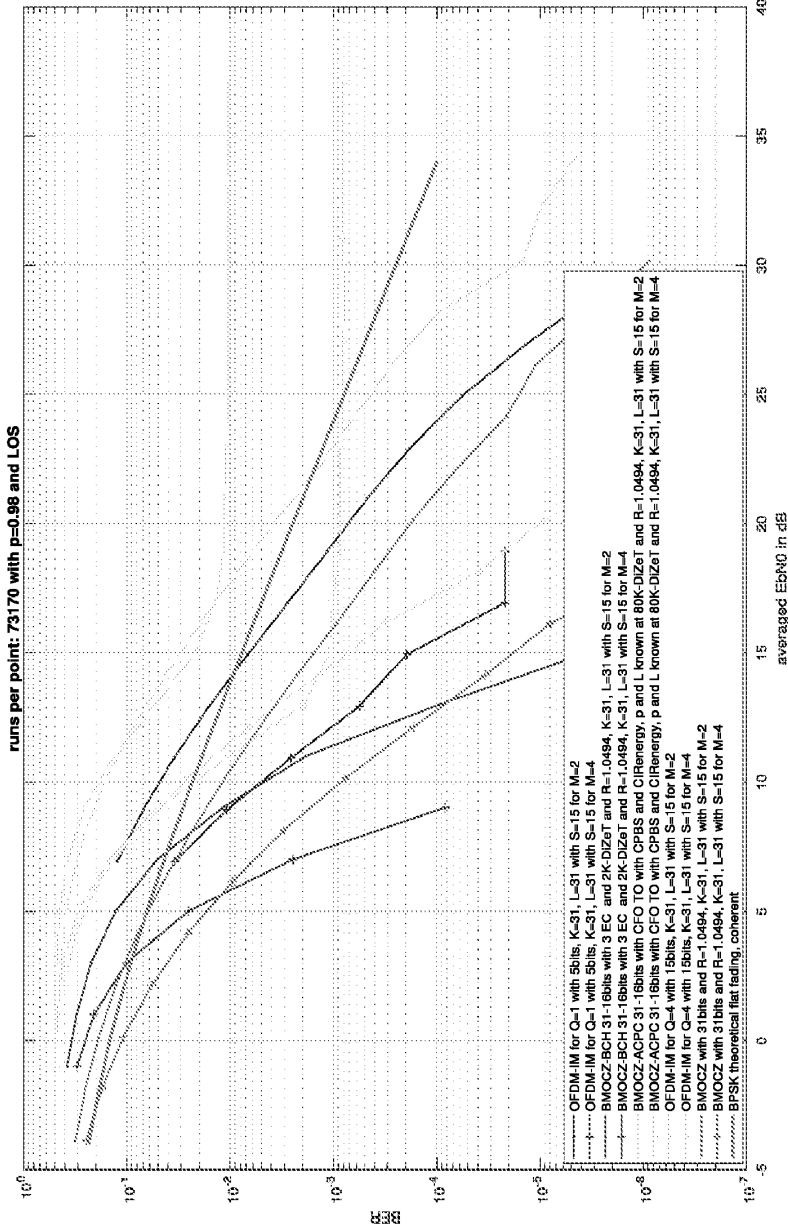


Fig. 13

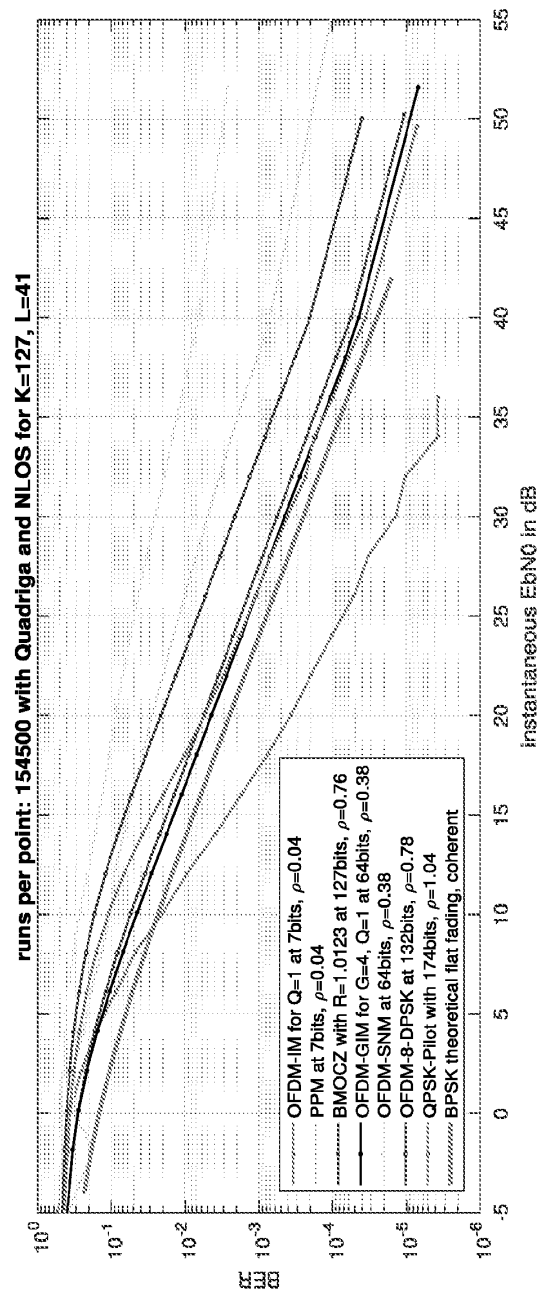


Fig. 14A

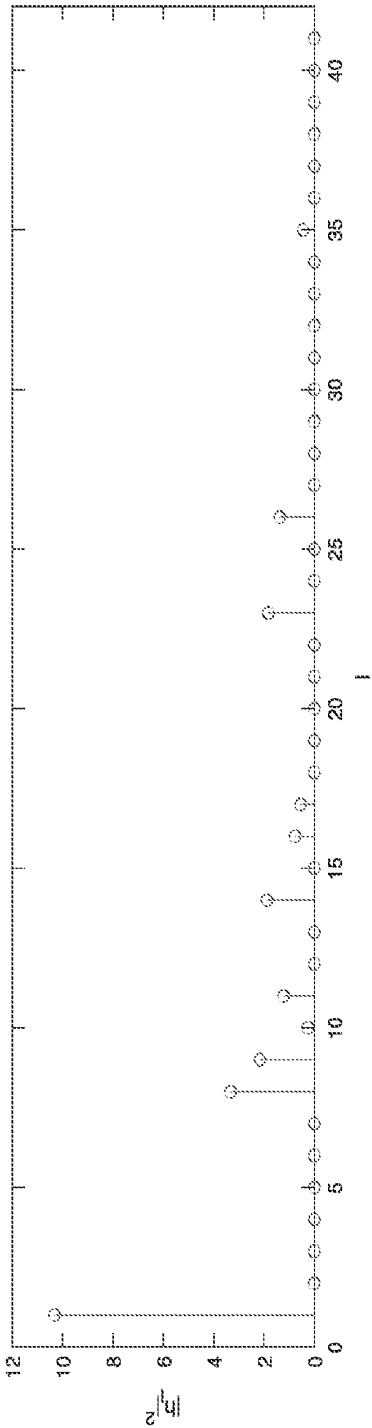


Fig. 14B

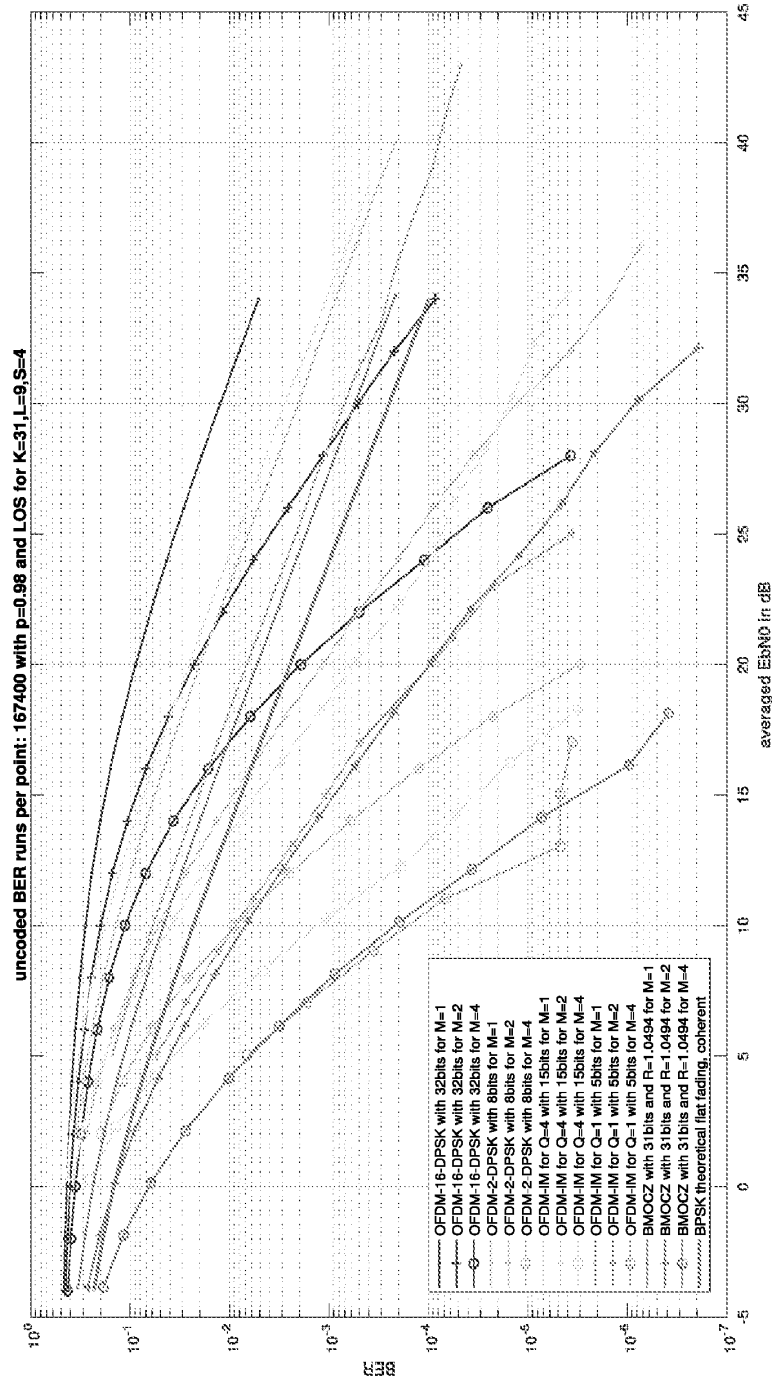


Fig. 15A

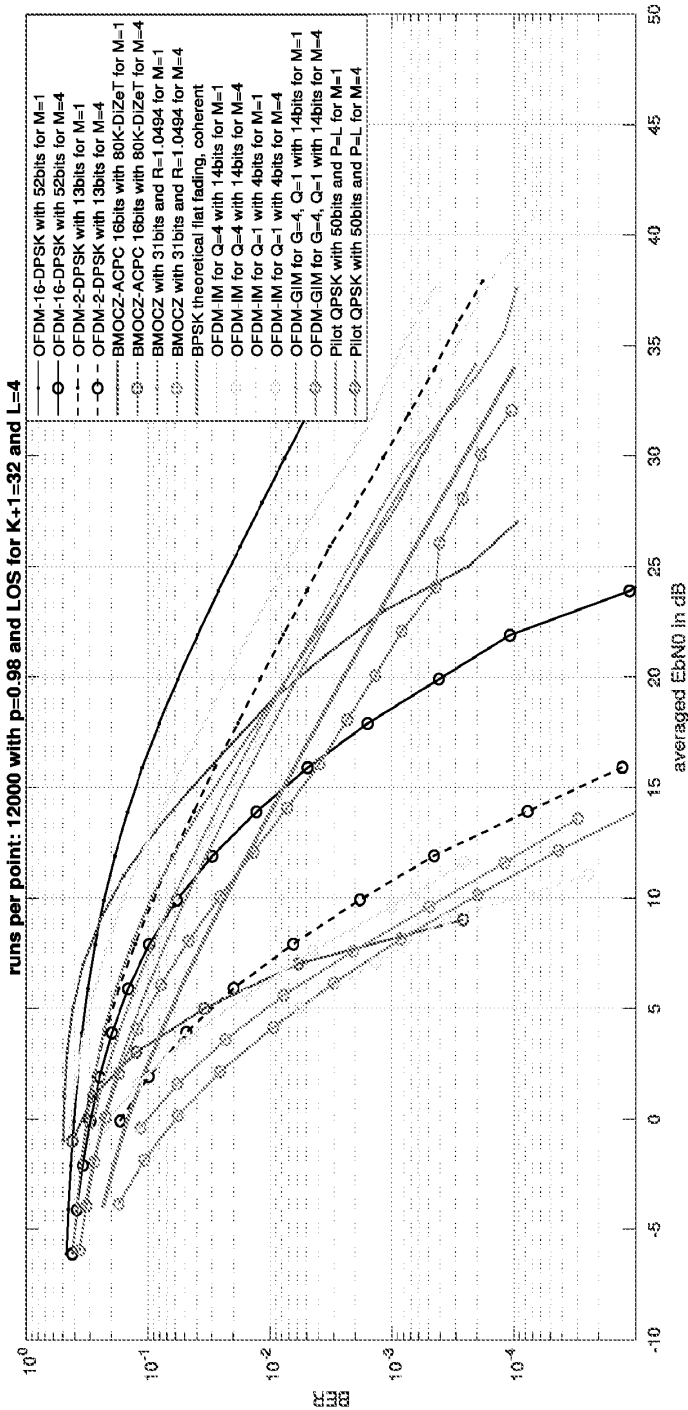


Fig. 15B

```

1  L=(n-k)/m; % Number of primitive ...
    polynomials for Gout
2  S=(n-1)/m;
3  B=2-m;
4
5  p=primpoly(m,'x');
6  gout=1; gin=deZhi(prim(1),n+1);
7  for s=1:S-1
8      gout=gfconv(deZhi(prim(s)),gout,2);
9  end
10 if L==numel(prim)
11     gin=1;
12 end
13 g=gfconv(gin,gout,2); % generator polynomial ...
    (Binary) for cyclic linear code
14 P=zeros(S,n-B); % Parity Check part
15 for i=1:B
16     xi=zeros(1,n+1-i); xi(n+1-i)=1;
17     [-,si]=gfdeconv(xi,g,2);
18     P(i+1,1:numel(si))=si;
19 end
20 tG=[P eye(B)]; % linear systematic generator ...
    matrix
21 tH=[eye(n-B) P']; % linear systematic generator ...
    matrix
22 a=[gout, zeros(1,n-m+1)]; % constant affine shift
23 P=zeros(K,n-k); % Parity Check part
24 for i=1:k
25     xi=zeros(1,n+1-i); xi(n+1-i)=1;
26     [-,si]=gfdeconv(xi,gout,2);
27     P(i+1,1:numel(si))=si;
28 end
29 tHout=[eye(n-k) P']; % used for syndrome, error ...
    correction
30 Tsynd=syndtable(tHout); % outer code syndrome table

```

Matlab-Code 1: Systematic Generator and Check Matrix

Fig. 16