

```
# onchain_monitor.py
```

```
import time
```

```
import requests
```

```
from config import PURCHASE_AMOUNT
```

```
from utils import analyze_liquidity,  
technical_analysis, execute_spot_order,  
send_email_notification
```

```
last_checked_symbols = set()
```

```
def get_new_onchain_tokens():
```

```
url = "https://api.bitget.com/api/v2/mix/  
market/coins-onchain"
```

```
try:
```

```
response = requests.get(url, timeout=5)
```

```
data = response.json()
```

```
return {item['symbol'] for item in  
data.get('data', [])}
```

```
except Exception as e:
```

```
print(f"[OnChain] فشل Bitget  
Onchain: {e}")
```

```
return set()
```

```
def monitor_onchain():
```

```
global last_checked_symbols
```

```
while True:
```

```
try:
```

```
current_symbols =  
get_new_onchain_tokens()
```

```
new_symbols = current_symbols -  
last_checked_symbols
```

```
for symbol in new_symbols:
```

```
    print(f"[OnChain] رمز جدید: {symbol}")
```

```
    if not analyze_liquidity(symbol,  
        onchain=True):
```

```
        continue
```

```
    if not technical_analysis(symbol,  
        onchain=True):
```

```
        continue
```

```
if execute_spot_order(symbol,  
PURCHASE_AMOUNT):
```

```
    send_email_notification(
```

```
        f"جدید OnChain شراء {symbol}",
```

```
        f"بمبلغ {symbol} تم شراء {PURCHASE_AMOUNT}$")
```

```
    last_checked_symbols = current_symbols
```

```
except Exception as e:
```

```
    print(f"[OnChain Monitor] خطأ: {e}")
```

```
    time.sleep(1)
```