

Integrating \$TSOTCHKE QRNG into your project on Solana mainnet:

[illegible]

```

    { pubkey: userTokenAccount.address, isSigner: false, isWritable: true },
    { pubkey: TREASURY_ADDRESS, isSigner: false, isWritable: true },
    { pubkey: TOKEN_PROGRAM_ID, isSigner: false, isWritable: false },
    { pubkey: configAddress, isSigner: false, isWritable: false },
    { pubkey: SYSVAR_CLOCK_PUBKEY, isSigner: false, isWritable: false },
  ],
  programId: PROGRAM_ID,
  data: instructionData,
});

// Send transaction
const signature = await sendAndConfirmTransaction(
  connection,
  transaction,
  [payer],
  { commitment: 'confirmed' }
);

return signature;
}

// Extract the random number from transaction
async function getRandomNumber(signature) {
  const txInfo = await connection.getTransaction(signature, {
    commitment: 'confirmed',
    maxSupportedTransactionVersion: 0
  });

  if (txInfo?.meta?.returnData) {
    const [encoded, encoding] = txInfo.meta.returnData;
    if (encoding === 'base64' && encoded) {
      const data = Buffer.from(encoded, 'base64');

      // Extract a uint64 random number
      const randomU64BigInt = data.readBigUInt64LE(0);
      return BigInt(randomU64BigInt);
    }
  }

  throw new Error('Failed to extract random number from transaction');
}

```

For more detailed integration instructions, visit our [GitHub repository](#).